Optimization and Supervised Machine Learning Methods for Inverse Design of Cellular Mechanical Metamaterials

Sheng Liu

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Mechanical Engineering

Pinar Acar, Chair

Bart Raeymaekers

Rakesh K. Kapania

Robert L. West

April 17, 2024

Blacksburg, Virginia

Keywords: Metamaterials, Genetic Algorithm, Deep Learning Method, Finite Element Analysis, Topology Optimization, Uncertainty Quantification

Optimization and Supervised Machine Learning Methods for Inverse Design of Cellular Mechanical Metamaterials

Sheng Liu

(ABSTRACT)

Cellular mechanical metamaterials (CMMs) are a special class of materials that consist of microstructural architectures of macroscopic hierarchical frameworks that can have extraordinary properties. These properties largely depend on the topology and arrangement of the unit cells constituting the microstructure. The material hierarchy facilitates the synthesis and design of CMMs on the micro-scale to achieve enhanced properties (i.e., improved strength, toughness, low density) on the component (macro)-scale. However, designing on-demand cellular metamaterials usually requires solving a challenging inverse problem to explore the complex structure-property relations. The first part of this study (Ch. 3) proposes an experience-free and systematic design methodology for microstructures of CMMs using an advanced stochastic searching algorithm called micro-genetic algorithm (μGA). Locally, this algorithm minimizes the computational expense of the genetic algorithm (GA) with a small population size and a conditionally reduced parameter space. Globally, the algorithm employs a new search strategy to avoid local convergence induced by the small population size and the complexity of the parameter space. What's more, inspired by natural evolution in the GA, this study applies the inverse design method with the standard GA (sGA) as a sampling algorithm for intuitively mapping material-property spaces of CMMs, which requires the selection of objective properties and stochastic search of property points within the property space. The mapping methodology utilizing the sGA is proposed in the second part of the study (Ch. 4). This methodology involves a robust strategy that is shown to identify

more comprehensive property spaces than traditional mapping approaches. The resulting property space allows designers to acknowledge the limitations of material performance, and select an appropriate class of CMMs based on the difficulty of the realization and fabrication of their microstructures. During the fabrication process, manufacturing defects cause uncertainty in the microstructures, and thus the structural properties. The third part of the study (Ch. 5) investigates the effects of the uncertainty stemming from manufacturing defects on the material property space. To accelerate the uncertainty quantification (UQ) via the Monte Carlo method, this study utilizes a machine learning technique to bypass the expensive simulations to compute properties. In addition to reducing the computational expense of the simulations, the deep learning method has been proven to be practical to accomplish non-intuitive design tasks. Due to the numerous combinations of properties and complex underlying geometries of metamaterials, it is numerically intractable to obtain optimal material designs that satisfy multiple user-defined performance criteria at the same time. Nevertheless, a deep learning method called conditional generative adversarial networks (CGANs) is capable of solving this many-to-many inverse problem. The fourth part of the study (Ch. 6) proposes a new inverse design framework using CGANs to overcome this challenge. Given combinations of target properties, the framework can generate a group of geometric patterns providing these target properties. Therefore, the proposed strategy provides alternative solutions to satisfy on-demand requirements while increasing the freedom in the fabrication process. Besides, with the advances in additive manufacturing (AM), the design space of an engineering material can be further enlarged by multi-scale topology optimization. As the interplay between microstructure and macrostructure drives the overall mechanical performance of engineering materials, it is necessary to develop a multi-scale design framework to optimize structural features in these two scales simultaneously. The final part of the study (Ch. 7) presents a concurrent multi-scale topology optimization method of CMMs. Structures in micro and macro scales are optimized concurrently by utilizing se-

quential quadratic programming (SQP) with the Solid Isotropic Material with Penalization (SIMP) method and a numerical homogenization approach.

Optimization and Supervised Machine Learning Methods for Inverse Design of Cellular Mechanical Metamaterials

Sheng Liu

(GENERAL AUDIENCE ABSTRACT)

Cellular materials widely exist in natural biological systems such as honeycombs, bones, and wood. Recent advances in additive manufacturing have enabled us to fabricate these materials with high precision. Inspired by architectures in nature, cellular mechanical metamaterials (CMMs) have been introduced recently as a new class of architected systems. The materials are formed by hierarchical microstructural topologies, which have a decisive influence on the structural performance at the macro-scale. Therefore, the design of these materials primarily focuses on the geometric arrangement of their microstructures rather than the chemical composition of their base material. Tailoring the microstructures of these materials can lead to several outstanding features, such as high stiffness and strength, low density, and high energy absorption. However, it is challenging to design microstructures that satisfy user-defined requirements for properties and material costs. This is mainly due to the trade-off between the accuracy and computing times of the optimization process. In the first part of this study (Ch. 3), a design framework is proposed to overcome this issue. The framework employs a global search algorithm called the genetic algorithm (GA). With a newly designed search algorithm, the framework reduces errors between target and optimized material properties while improving computational efficiency. Inspired by the algorithm behind the GA, the second part of the study (Ch. 4) employs a similar algorithm to identify a material property chart demonstrating all possible combinations of mechanical properties of CMMs. Each axis of the material property chart corresponds to a selected

mechanical property, such as Young's modulus or Poisson's ratio, along different directions. The boundary of the property space helps designers understand material performance limitations and make informed decisions in engineering practices. In the fabrication process, unexpected material properties might be achieved due to defects and tolerances in additive manufacturing (AM), such as uneven surfaces, shrinkage of pores, etc. The third part of the study (Ch. 5) investigates the uncertainty propagation on mechanical properties as a result of these manufacturing defects. To investigate the uncertainty propagation problem efficiently, the study uses a deep learning method to predict the variations (stochasticity) of properties. Consequently, the material property space boundary also varies with the uncertainty of properties. In addition to their computational efficiency, deep learning methods are beneficial for solving many-to-many inverse design problems. Traditionally, the global and local search/optimization methods retrieve alternative optimal solutions in their Pareto front set, where each solution is considered to be equally good. A deep learning method called conditional generative adversarial networks (CGANs) can bypass the property calculation to accelerate the simulation process while obtaining a group of candidates with on-demand properties. The fourth part of the study (Ch. 6) employs CGANs to build a new inverse design framework to increase flexibility in the fabrication process by generating alternative solutions for the microstructures of CMMs. Besides, as fabrication technologies have advanced, designing engineering systems has become increasingly complex. Material design is now not only focused on meeting micro-scale requirements but also addressing needs at multiple scales. The interaction between the microstructure (small-scale) and macrostructure (large-scale) significantly influences the overall performance of engineering systems. To optimize structures effectively, there is a need for a design framework that considers these two scales simultaneously. Thus, the final part of the study (Ch. 7) introduces a method called concurrent multi-scale topology optimization. To obtain the extreme performance of a multi-scale structure, this approach optimizes its structure at both micro- and macro-scales

concurrently, using gradient-based optimization algorithms with density-based property determination methods in the two scales.

Dedication

To my family

Acknowledgments

As time flies by, I find myself on the four-year academic journey at Virginia Tech filled with heartfelt gratitude. In this wonderful journey, I have been fortunate to have my dedicated mentor, Professor Pinar Acar. Words cannot express my deepest thanks to my professor. Under her invaluable guidance, I have felt the charm of academic research and understood the importance of diligent effort. Her inspiration and encouragement have kept me persevering through challenges, instilling in me the belief that I can overcome any obstacle. Beyond academic guidance, I appreciate her concern for my life and career aspirations. Her advice has provided clarity about my interests and potential, motivating me to pursue my dreams bravely.

It is my greatest honor to have Dr. Bart Raeymaekers, Dr. Rakesh Kapania, and Dr. Robert West on my thesis committee. I would like to express my deepest gratitude for their scholarly advice and insightful feedback that greatly improved the quality of my dissertation. I would also like to thank all my colleagues in the Advanced Structure and Optimization (ASTRO) lab: Arulmurugan Senthilnathan, Mahmudul Hasan, Ender Eger, Maruf Billah, Waris Khan, Kiara McMillan, Mohamed Elleithy, and Matthew Long for their stimulating discussions and incisive comments.

My sincere appreciation goes to my family for providing spiritual support throughout the entirety of my life. Their love and care have served as an enduring wellspring of strength and motivation for me.

Finally, I wish to acknowledge the financial support from the National Science Foundation for providing the resources and facilities necessary to carry out this research.

Contents

List of Figures

List of Tables

List of Abbreviations

$2D$	Two dimensions
$3D$	Three dimensions
$8D$	Eight dimensions
%diff	Percentage differences between deterministic properties and properties under tolerance cases
α	Step size
β	Constant wave number
Δ	Step size of the finite difference approximation
γ_i	Phase angle
$\hat{\sigma}$	Volume-averaged stress
$\hat{\varepsilon}$	Volume-averaged strain
$\hat{C}_{ij}$	Second-order homogenized elastic modulus tensor
$\hat{e}$	Cartesian basis
$\hat{H}_j$	Convolution operator of the sensitivity filter
$\hat{y}_i$	Predicted properties of the materials
λ	Lagrange multipliers of the Karush–Kuhn–Tucker conditions

μGA	Micro-genetic algorithm
ν	Poisson's ratio
ω	Parameter sets of neurons of the hidden layers
$\overline{x}_{prop}$	Mean values of the structural properties
ϕ	Compliance
ψ	Local phase field
ψ_0	Threshold
σ	Stress
$\sigma_{x_{prop}}$	Variances of the structural properties
θ	Polar angle
$\tilde{w}$	Weighted coefficient
ε	Strain
φ	Porosity
ζ	Geometric variables of porous shapes
A	Active constraint matrix
B	B matrix of the element
C_i	Side lengths of reduced parameter space
D	Discriminator
d	Search direction

$d_{boundary,x}$ Horizontal distances of $d_{shortest}$

$d_{boundary,y}$ Vertical distances of $d_{shortest}$

$d_{shortest}$ Shortest Euclidean distance between objective points of GA and property points in the material property space

d_{target} Shortest Euclidean distance between objective points on the material property space boundary

$d_{validation}$ Shortest Euclidean distance between objective points of GA for validating the maxial material property space

$dist(e, f)$ Distance between centers of the elements in the macro-scale

E_e Stiffness approximated by the SIMP approach

E_0 Stiffness of the base material

E_{ij} Young's modulus

E_{min} Stiffness of void sections in the multi-scale structure

F External force applied on the macrostructure

f_v Volume fraction of the multi-scale structure

f_{best} Best objective function value

G Generator

G_{ij} Shear modulus

K Global stiffness matrix

k_e	stiffness matrix of the element
K_{best}	Fittest individual
L	Side length
L_D	Loss function of the discriminator
L_G	Loss function of the generator
L_{tolr}	Tolerance of the side length
N_P	Population size of genetic algrithm
$nelx$	Number of elements in the longitudinal direction of the design domain
$nely$	Number of elements in the transverse direction of the design domain
P	Population
p	Penalty factor of the SIMP approach
R^2	Coefficient of determination
r_{min}	Radius of the sensitivity filter
S	Independent solver
t	Surface traction
U	Global displacement vector
u	Local/nodal displacement
V	Volume of RVEs
X	Presence of the materials in the spinodal topologies

x_{prop} Structural properties

y_i Actual properties of the materials

z Random noise vector

$x_{prop,95\%}$ 95% confidence interval of the structural properties under uncertainty

AM Additive manufacturing

ANBI Adaptive Normal Boundary Intersection

ASTRO Advanced Structure and Optimization

AWS Adaptive Weighted Sum

B Search ball of the ESS

BCE Binary cross-entropy

BFGS Broyden–Fletcher–Goldfarb–Shanno

CAE Computer-aided engineering

CGANs Conditional generative adversarial networks

CMMs Cellular mechanical metamaterials

CNN convolutional neural network

DDR4 Double Data Rate fourth generation

DNNs Deep neural networks

EBHM Energy-based homogenization method

ESS Empirical empty space statistics

FD Finite difference

FDM Fused Deposition Modeling

FE Finite element

FEM Finite element method

FFF Fused Filament Fabrication

GA Genetic algorithm

GAN Generative adversarial network

GRF Gaussian random field

GWS Generalized Weighted Sum

KKT Karush–Kuhn–Tucker

LHS Latin hypercube sampling

MBB Messerschmitt-Bölkow-Blohm

MC Monte Carlo

ML Machine learning

MLP Multi-layer Perceptron

MMA Method of moving asymptotes

MSE Mean squared error

NNs Neural networks

NSG Niederreither sequence generator

PBC Periodical boundary condition

PEI Polyetherimide

PLA Polylactic acid

Pr Probability

PRNG Pseudo-random number generator

PUC Periodical unit cells

ReLU Rectified linear unit

RVEs Representative volume elements

sGA Standard genetic algorithm

SIMP Solid Isotropic Material with Penalization

SQP Sequential quadratic programming

SSG Sobol sequence generator

TO Topology optimization

UHMWPE Ultra-high molecular weight polyethylene

ULTEM Branded name for Polyetherimide

UQ Uncertainty quantification

.

Chapter 1

INTRODUCTION

Cellular structures have become one of the most promising material systems owing to their high strength and lightweight nature [1]. With their unique features, these materials have been used in widespread applications of aerospace [2, 3] and transportation [4, 5] industries, such as in structural panels. In recent years, the advent of additive manufacturing has allowed the realization of high-performance and high-precision industrial parts and products built with cellular mechanical metamaterials (CMMs). The CMMs are assembled with representative volume elements (RVEs). The effective mechanical properties of CMMs depend more on the geometric patterns of RVEs than the properties of the constitutive (base) material [6, 7]. Consequently, the design of cellular materials is mainly studied by optimizing the geometric patterns of RVEs, which are distributed within their design domain. To meet various industrial and academic requirements and realize the corresponding metamaterials, there is a need for a systematic design framework to exploit the implicit relationship between the geometric patterns of RVEs and their corresponding properties. Chapter 1 discusses the potential issues in the design of cellular materials, which motivate us to study the following five topics: exploring their complex parameter space (1.1.1), investigating their material property space (1.1.2), quantifying uncertainty in their properties (1.1.3), solving their many-to-many problems (1.1.4), and analyzing their multi-scale structures (1.1.5). The contribution of the above studies is summarized in (1.2).

1.1 Motivation

1.1.1 Parameter space exploration of porous metamaterials

Parameterization of material microstructures enables a designer to tune geometric variables for specific industry applications corresponding to particular material properties, such as Young's modulus, Poisson's ratio, porosity, etc. Traditionally, investigating the optimal micro-structures requires repeated attempts to combine the geometric parameters. Without any prior information, the search is time-consuming due to complex simulations or experimental approaches to evaluate the structural responses of every test micro-structure [8, 9]. Therefore, with the increased complexity of materials design, the design strategy must have an effective method for finding a feasible set of parameter values that define the micro-structures of new materials. Recently, advances in deep learning have been beneficial for the design of new materials using various neural networks (NNs)[10]. However, there are several potential issues arising from using NNs for materials design: (i) high computational expenses of building an extensive database for structure-property relations [11], (ii) number of trails and tests in constructing architectures of NNs to avoid overfitting issues while maintaining high accuracy, and (iii) difficulties in interpreting the physical meaning of the connection between the hidden neurons of NNs [12].

Unlike deep learning, genetic algorithm (GA) is independent of prior knowledge about the structure-property relations and searches for the optimum solutions among realistic micro-structures of RVEs. Moreover, the micro GA (μGA) boasts the computational efficiency of the standard GA (sGA) as it only needs a small population. However, the small population size in the μGA induces the low diversity of the candidates in each iterative optimization process, leading to premature convergence to the local optimum. Therefore, it is necessary to develop an advanced search strategy to overcome the local convergence. Besides, the works

of Maaranen et al. [13, 14] indicate that initial populations may significantly influence the optimization results in the GA over several generations. In addition to pseudo-random numbers applied in traditional GA, it is worth studying several alternative ways to generate the initial population of the GA within the parameter space, which might improve the performance of the GA as well. Consequently, with the μGA improved by an advanced search strategy, we can outline a GA-based design framework for parameterized CMMs that is independent of prior knowledge about structure-property relations and allows users to interpret physical meaning during optimization. Figure 1.1 illustrates the design framework of the parameterized CMMs.

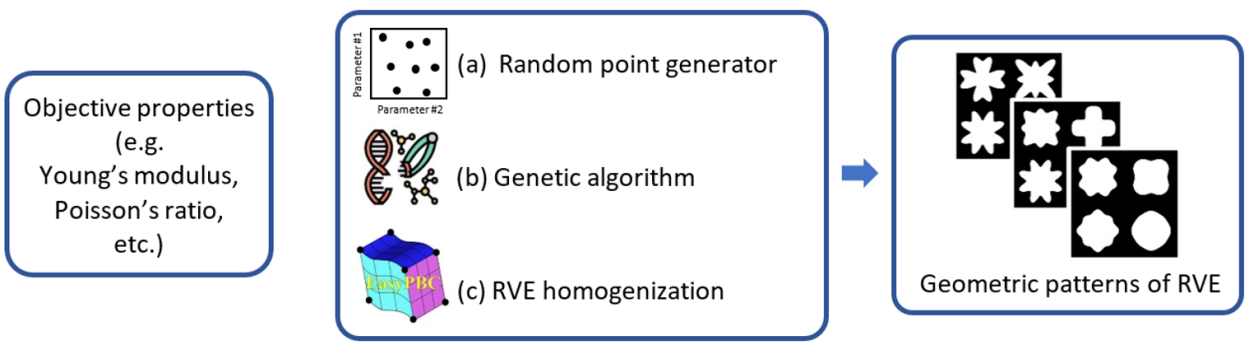


Figure 1.1: Schematic for the presented inverse design approach for CMMs (GA and RVE homogenization figures courtesy of [15, 16]).

1.1.2 Material-property space exploration of porous metamaterials

Material property space serves as a valuable tool/reference for material selection. Similar to Ashby's chart [17], it is a complete envelope that summarizes theoretically possible combinations of properties, denoted as experimental points in the space. Its area represents the availability of the combined performance of a particular family of metamaterials. Referring to their corresponding areas, engineers can check the limitations of the combined perfor-

mances of materials and compare them from one family to another to intuitively select an appropriate material. Methodologies of mapping the property space can be classified into analytical and computational methods. Analytical methods generate the property space by using bounding theories such as the Hashin-Shtrikman bounds [18], and the Voigt [19] and Reuss [20] bounds. These theories predict points that are exterior to the bounds to expand the property space. However, the points are non-physical and might not correspond to physically realizable microstructures. What's more, the bound determined by these theories is for a single property instead of simultaneous combinations of properties. Some computational methods utilize uniform sampling of geometric design variables to map the property space. Although it is a rapid method to sample numerous possible properties corresponding to their randomly generated microstructures, it produces a poor approximation of the property space with a small region and centrally located property points due to the central limit theorem. Pareto Front techniques can provide more accurate estimations of the property space [21, 22, 23]. The drawback is relying on appropriately selected initial guesses to process gradient-based optimization methods [24]. Their optimum solution might converge to the local solutions considering the complexity of the design space caused by complicated geometric arrangements of metamaterials.

GA traditionally serves as a global optimization method to search extreme properties in forward problems [25, 26, 27] or to obtain optimal microstructures with target properties in inverse problems [28, 29, 30]. In these problems, after a stochastic search process of the GA, the engineers are usually interested in its final optimal solutions rather than its children generated in each generation. On the contrary, the principle of natural evolution in the GA can help us intuitively sample possible microstructures (children of the GA) for a particular material system. After computing the structural properties of these microstructures, the material-property space region can be determined by bounding all of the resulting property

points distributed in a 2D material property chart. Moreover, GA is independent of the initial guess and globally searches solutions within the design domain. Hence, inspired by natural evolution, we can develop a GA-based mapping framework to systematically explore the material-property space of parametrized CMMs while addressing the potential issues mentioned above. The flowchart of mapping the material-property space of parametrized CMMs is shown in Fig. 1.2.

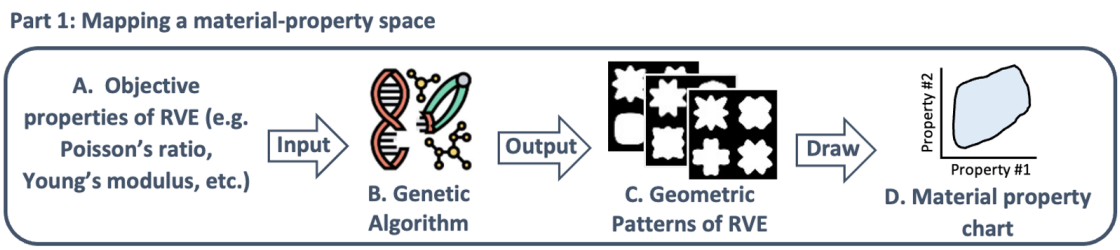


Figure 1.2: Schematics of the presented procedures to map a deterministic material-property space (GA algorithm figure courtesy of [15]).

1.1.3 Material-property space exploration of porous metamaterials under uncertainty

Apart from the material selection, the realization of the CMMs introduces another challenge in the material study. Compared with conventional non-architected materials, the CMMs consist of microstructures characterized by more complicated geometry patterns of RVEs. Their complex underlying structures cause their realization processes, such as Fused Deposition Modeling (FDM) and Fused Filament Fabrication (FFF) [31], to suffer from anomalies related to machining tolerances, surface roughness, and other inconsistencies [31, 32]. For example, surface distortion appears for thermoplastic materials caused by various cooling rates and complex porous microstructures in RVEs [32] during the cooling process of FFF/FDM. Due to the unique features of parameterized CMMs, slight variations in their values

enable us to approximate the realization of this manufacturing defect.

Monte Carlo (MC) simulation is a traditional method of quantifying uncertainty. It helps us understand the statistics of stochastic data points by running thousands or millions of virtual experiments. However, running a significant number of virtual experiments requires substantial computational power and time, especially when dealing with expensive finite element analysis in the property calculation of CMMs. Recent advances in machine learning methods are beneficial to exploring property-structure relations by building NN-based surrogate models. A well-trained surrogate model can effectively predict the structural properties of various geometric patterns of the RVEs realized by slight variations in the geometric parameters. It accelerates the MC simulation by bypassing expensive computations on properties. Following up on the study in Sec. 1.1.2, a new mapping framework is required to visualize the effect of manufacturing defects on the region of the material property space of CMMs. With the MC simulation facilitated by the deep learning method, uncertainty in material property space can be efficiently quantified and intuitively visualized as a variation interval of the space boundary. Figure 2.1 summarizes the flowchart of mapping the material-property space of parameterized CMMs under uncertainty.

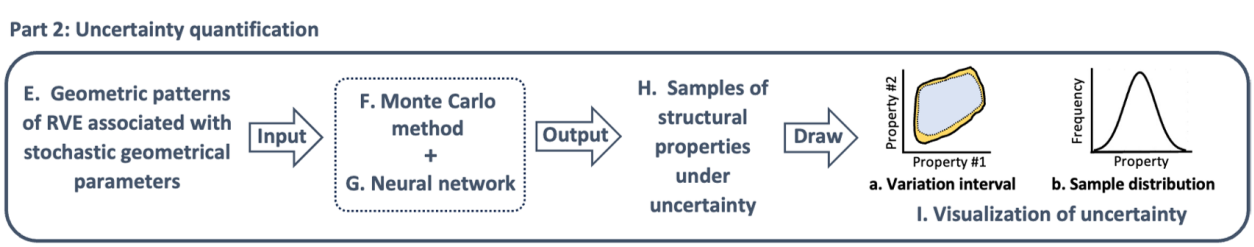


Figure 1.3: Schematics of the presented procedures to map a stochastic material-property space.

1.1.4 Generative adversarial networks for the inverse design of spinodoid metamaterials

The traditional metamaterials involving microstructures defined as trusses, jointed connections, and sharp edges are subject to high-stress concentrations, leading to poor recoverability and unexpected mechanical behavior [33]. To overcome potential issues due to high-stress concentrations while maintaining outstanding mechanical performance, Kumar et al. [33] reported a new class of metamaterials formed by spinodal topologies called spinodoid metamaterials. These materials are naturally generated by a phase separation process known as spinodal decomposition [33]. Spinodoid topologies involve materials with continuous and smooth surfaces having approximately zero-mean curvature in their microstructures [33]. This makes spinodoid metamaterials less vulnerable and more resistant to structural failure. Unlike the previously mentioned metamaterials, spinodoid metamaterials are non-periodic, enlarging their design space and contributing to the comprehensive and seamless range of anisotropic mechanical properties [33]. Moreover, spinodoid metamaterials can exhibit a negative Poisson's ratio, which is advantageous for aerospace systems, such as deployable and morphing structures. In addition to the industrial applications, tailorable anisotropic elastic modulus allows spinodoid metamaterials to be applied in the medical field, e.g., patient-specific bone replacements by reproducing trabecular bone properties [33].

The critical issue in designing spinodoid metamaterials is retrieving their internal geometrical arrangements for user-defined mechanical performance. The recent developments in machine learning techniques have demonstrated a significant advantage in solving challenges faced in the inverse design of spinodoid metamaterials. In the work of Kumar et al. [33], a multiple-layer perception (MLP) trained on spinodoid metamaterials bypasses expensive simulations and experiments and efficiently predicts an optimal topology for a prescribed set

of mechanical properties while avoiding ill-posed inverse problems. In the work of Roding et al. [34], an appropriate Bayesian computation framework, facilitated by convolutional neural network and Gaussian random field, accelerates exploration of design space and predicts diffusivity of spinodoid metamaterials in all three directions. Although these recent studies have proposed several novel training approaches for spinodoid metamaterials for different engineering purposes, the many-to-many inverse problem is not addressed. The problem in material design is finding a group of candidate geometric patterns of microstructures given a prescribed combination of on-demand effective properties. Referring to Kumar's work [33], multiple topologies of spinodoid metamaterials can contribute to identical or similar effective properties. This promotes freedom in actual fabrication by providing alternative solutions that fit manufacturing requirements. Therefore, solving the many-to-many inverse problem is important to the material design. It requires proposing an inverse design framework of 2D spinodoid metamaterials by employing a conditional generative adversarial network (CGAN) that enables the generation of a group of candidate geometric patterns with sets of target effective properties. The flowchart of the design framework is represented in Fig. 1.4.

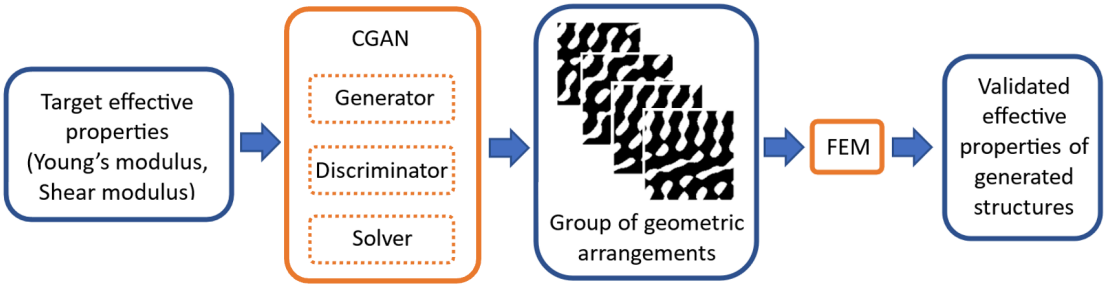


Figure 1.4: Schematic of the inverse design framework of spinodoid metamaterials.

1.1.5 Concurrent multi-scale topology optimization of porous metamaterials

With the development of micro-scale additive manufacturing, innovative materials with complex microstructures can be easily manufactured. The advanced technique reduces the characteristic sizes of engineered structures and promotes freedom in engineering design to meet requirements from academic and industrial fields. Multi-scale structures are favored by engineers due to their exceptional designability and outstanding performance, such as low density, high stiffness, high energy absorption, and multi-functionality [35, 36, 37, 38]. Commonly, there are two or more characteristic scales consisting of multi-scale structures. On the micro-scale, the distribution and geometric patterns of microstructures affect the performance of macrostructures. On the macro-scale, the external boundary conditions affect structural design in the two scales. Therefore, in a multi-scale structure, macrostructures interplay with microstructures to dominate the overall performance of the multi-scale structure. Topology optimization (TO) is a computational process to achieve the desired performance of the physical system by optimizing structural layouts in a given design domain, boundary conditions, and constraints [39]. Owing to its simplicity, the solid isotropic material with penalization (SIMP) method has become a popular mathematical approach for TO. As a starting point, Sigmund [40] has developed a 99-line MATLAB code for TO with the SIMP method. This opens an avenue for density-based TO methods. After this, the 88-line code has been proposed by Andreassen et al. [41]. It can be considered a compact version of the 99-line code, which increases computational efficiency and offers several alternative density filters. Unlike the previous two codes that employ the optimality criteria, Liu et al. [42] has presented a MATLAB code to solve TO problems with non-linear programming, such as sequential quadratic programming (SQP) and method of moving asymptotes (MMA). Non-linear programming is beneficial for quadratic problems and is more accurate than linear

programming, such as the optimality criteria methods applied in the works of Sigmund [40] and Andreassen et al.[41]. By referring to the 88-line code, Gao et al. [43] have introduced a concurrent TO framework developed as a MATLAB code by integrating the SIMP and homogenization methods to evaluate structural performance in two scales.

From the above studies, we can take advantage of the works of Liu et al. [42] and Gao et al. [43] to develop a concurrent multi-scale TO framework for CMMs. The proposed framework aims to optimize structural layout simultaneously in two scales to obtain minimum compliance of a multi-scale structure with a given boundary condition and volume fraction. The flowchart of the TO framework is shown in Fig. 1.5. At the micro-scale, the microstructures of CMMs are optimized based on the energy-based homogenization (EBHM) method. At the macro scale, the simplicity of the SIMP method promotes intuitive and compact layouts of the MATLAB code to estimate the properties of discretized macrostructures. The design variables of the TO problem are updated by using SQP due to the existence of highly non-linear implicit terms (homogenized elastic modulus tensor of CMMs) in the objective function. The SQP can be performed using a gradient-based MATLAB optimization toolbox called *fmincon*. With default settings, the Hessian matrix of the objective function is usually approximated by means of a Quasi-Newtown method called Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [42]. However, by considering the complex parameter space of the microstructure [44], the homogenized tensor of the material might be sensitive to variations in its geometric design variables. To maintain the stability of the sensitivities of the tensor, the Hessian matrix in the framework is numerically calculated using Hooke's law and partial differential equations in each sub-quadratic problem of the SQP.

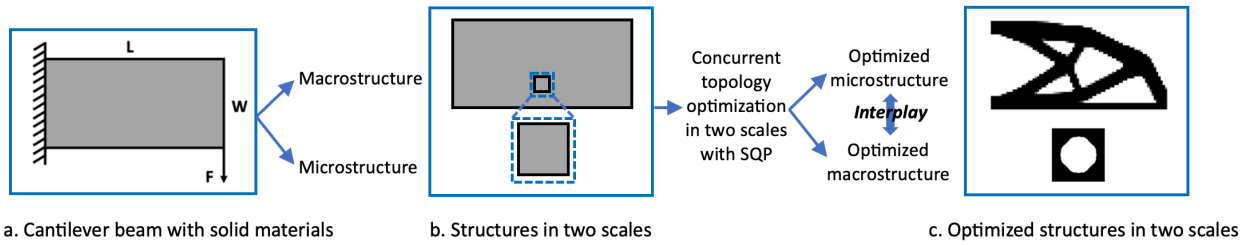


Figure 1.5: Schematic of the concurrent multiscale design framework of porous metamaterials.

1.2 Contributions

The overall contributions of this study in the fields of multi-scale inverse design, uncertainty quantification, and data-driven modeling of CMMs are summarized below:

- Investigation of the **parameter space** of porous metamaterials.
- Investigation of the **theoretical limits** of mechanical properties of porous metamaterials (**material property space**).
- Investigation of the effects of fabrication/processing-related **uncertainty on the material property space**.
- Inverse design of **structure-property linkages** of anisotropic mechanical properties of spinodoid metamaterials with data-driven surrogate models.
- Investigation of **multi-scale structures** built with porous metamaterials.

Chapter 2

MATHEMATICAL BACKGROUND

This chapter discusses the mathematical background for the inverse design of cellular mechanical metamaterials (CMMs) by introducing formulations to generate CMM geometries, determine their homogenized elastic properties, optimize their microstructures with objective properties, and build relationships between microstructures and their corresponding properties. The study primarily focuses on two classes of CMMs: porous and spinodoid metamaterials. The characterization of their microstructures is defined in Sec. 2.1. Based on the microstructures, their homogenized elastic properties can be determined by the homogenization methods in Sec. 2.2. Section 2.3 introduces gradient-based and gradient-free optimization methods to solve inverse and topology optimization (TO) problems of CMMs parameterized by their geometric variables. Advances in machine learning allow us to explore the implicit relationship between structures and properties of CMMs, and accelerate computational processes in materials design by bypassing the expensive simulations for property determination. Section 2.4 lists all the supervised machine learning models applied in the study.

2.1 Computational Characterization of Metamaterials

The study mainly focuses on two types of CMMs, porous and spinodoid metamaterials. Their effective properties are determined by the arrangement and geometry patterns of their

microstructures called RVEs. Section 2.1 provides mathematical formulations in characterizing the geometry patterns of RVEs for two classes of metamaterials in the study. The size of RVEs should be not only sufficiently large to avoid introducing non-existing properties but also small enough to maintain the computational efficiency [45].

2.1.1 Porous metamaterials

A special but widely used class of CMMs is called porous metamaterials. These CMMs contain periodically repeating RVEs. Their mechanical properties mainly depend on the geometry of the underlying unit cells, such as the shape of the pore as shown in Fig. 2.1. In unit cells of these CMMs, the contours of the pores are four-fold symmetrical and defined as [46]:

$$r(\theta) = r_o(1 + \zeta_1 cos(4\theta) + \zeta_2 cos(8\theta)) \tag{2.1}$$

$$r_o = L\sqrt{\frac{2\phi}{\pi(\zeta_1^2 + \zeta_2^2 + 2)}} \tag{2.2}$$

where θ is the polar angle with $0 \leq \theta \leq 2\pi$, r is the polar radius, L is the size of the unit cell, and φ is the porosity. The geometric parameters $\zeta = (\zeta_1,\zeta_2)$ determine the family of different porous shapes. Fig. 2.1 demonstrates several sample geometric patterns of unit cells with various pore shapes controlled by the parameters ζ. In this work, L and φ are assumed to be constant and equal to 10 mm and 0.45, respectively.

The simplest RVEs of these CMMs consist of periodic unit cells (PUCs) shown in Fig. 2.2(a). Their geometry pattern is simply determined by two geometric parameters ζ_i (i = 1,2). In Fig. 2.2(b), the complexity of geometry patterns of RVEs can be increased by introducing

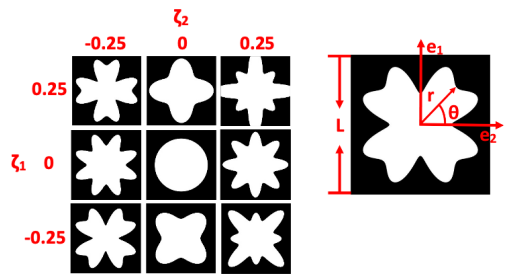


Figure 2.1: Sample porous shapes generated by the formulations given in Eqs. 2.1 and 2.2.

RVEs with non-periodic unit cells. As the number of unique unit cells in the RVE becomes as large as $n \times n$, the number of geometric parameters ζ_i increases to $2 \times n \times n$, potentially leading to an increased level of variability in geometric dimensions and corresponding properties of the CMMs. Each pair of geometric parameters (ζ) controls the associated contour of the particular unit cell in the RVE. All the geometric parameters are assigned in order as a geometric matrix shown in Fig. 2.2.

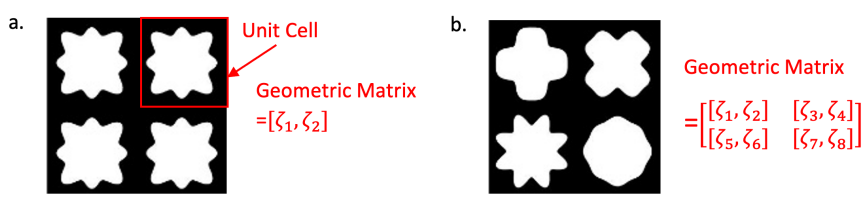


Figure 2.2: RVEs of the CMMs with (a) PUCs and (b) non-PUCs.

2.1.2 Spinodoid metamaterials

A new class of metamaterials formed by spinodal topologies has been developed in recent years and called as spinodoid metamaterials. These materials are obtained by simulating spinodal decomposition during the early phase separation stage, where homogeneous material decomposes into two different material phases [47]. The Cahn-Hilliard model describes the phase separation, representing the concentration fluctuation of one local phase field $\psi(x)$. Mathematically, the phase field can also be described by a Gaussian random field (GRF),

as a superposition of standing waves illustrated as given by Kumar et al. [33]:

$$\psi(x) = \sqrt{\frac{2}{N}} \sum_{i=1}^{N>>1} \cos(\beta n_i \cdot x + \gamma_i). \tag{2.3}$$

Here, N is the number of the cosine waves, β is a constant wave number that is larger than zero, n_i is the uniformly distributed direction, and γ_i is the phase angle of the ith wave vector. By aligning the GRF with Cartesian basis $\{\hat{e_1}, \hat{e_2}\}$, a resulting anisotropic topology of 2D spinodoid metamaterials is approximately represented in a non-uniform orientation distribution function described by [33]:

$$n_i \sim U(\{k \in \Omega : (|k \cdot \hat{e_1}| > \cos\theta_1) \oplus (|k \cdot \hat{e_2}| > \cos\theta_2)\}), \tag{2.4}$$

where θ_1 and θ_2 are angles in the Cartesian coordinate system in Fig. 2.3(a). The angles mainly control the resulting spinodal topologies, and they are within $\theta \cup \{\theta_{min}, \frac{\pi}{2}\}$. Ω denotes a circle of unit radius in Fig. 2.3(a). Soyarslan et al. [48] introduced a level-set ψ_0 to generate bi-continuous solid–void microstructures by using a binary indicator function defined as:

$$X(x) = \begin{cases} 1 \ \text{(solid section)}, & \text{if } \psi(x) \leq \psi_0 \\ 0 \ \text{(void section)}, & \text{if } \psi(x) > \psi_0 \end{cases} \tag{2.5}$$

where $X(x)$ denotes the presence of the materials in the spinodal topologies. The binary indicator function defines the two separated phases as void and solid sections based on the threshold ϕ_0. The threshold value is related to porosity ϕ_0 of the microstructures [33]. This study considers a 2D RVE for spinodoid microstructures generated by the binary indicator function. Figure 2.3 shows the geometric patterns of the 2D RVEs at $\phi_0 = 0.35$. A complete set of design variables $\{\theta_1, \theta_2, \phi_0\}$ characterizes 2D RVEs of the spinodoid metamaterials.

This study chooses $\theta_{min} = 15°$ [33] and $\phi_0 = 0.35$ to guarantee that solid domains of the RVE have good structural connectivity.

Figure 2.3: Spinodoid topologies: (a) Schematic of the geometrical parameters: θ_1, θ_2; (b) Sample microstructures of 2D spinodoid metamaterials at $\phi_0 = 0.35$ in 128×128 pixel resolution.

2.2 Computation of Elastic Properties

In the scope of linear elastic materials, the homogenization method is applied to evaluate the equivalent constitutive mechanical behavior of microstructures (geometric patterns of RVEs), called homogenized structural properties. The basic idea of the homogenization method is to apply unit tensile strain on a finite element model at the RVE level with user-defined boundary conditions. The unit strain tests are obtained by the following numerical experiments: one unit axial stretch imposed along each of two principal axes (Fig. 2.4a and 2.4b) and one unit shear strain imposed along both axes (Fig. 2.4c) [16].

2.2.1 Energy-based homogenization method

The traditional homogenization method with asymptotic expansion theory has complex theoretical derivations and numerical implements. Therefore, it is hard to directly connect this algorithm with the numerical homogenization method and topology optimization for ma-

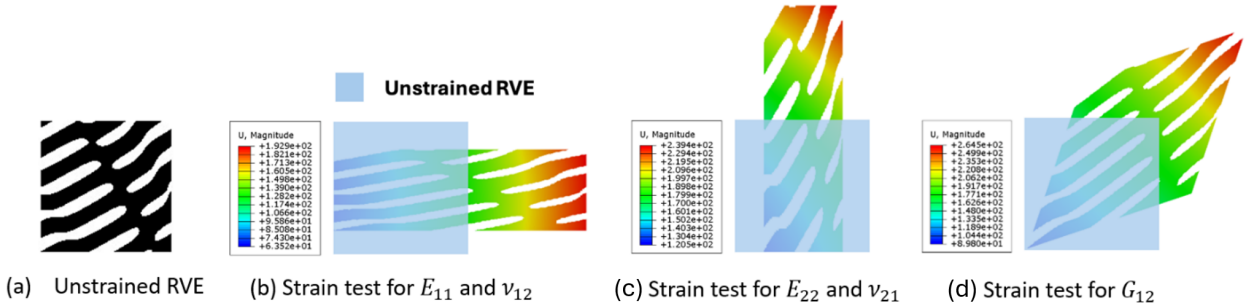


Figure 2.4: Strain tests: (a) Unstrained RVE; Deformed RVEs under strain tests for (b) E_{11}, (c) E_{22}, and (d) G_{12}.

terial design [49]. To overcome the above issue, the energy-based homogenization method (EBHM) has been proposed in recent studies [43, 50, 51]. The EBHM employs the average stress and strain theorems to evaluate the element mutual energies of discretized RVEs, as given in the following equations:

$$\overline{\sigma}_{ij} = \frac{1}{V_{RVE}} \int_V \sigma_{ij} dV \tag{2.6}$$

$$\overline{\varepsilon}_{ij} = \frac{1}{V_{RVE}} \int_V \varepsilon_{ij} dV \tag{2.7}$$

where $\overline{\sigma}$ and $\overline{\varepsilon}$ denote the volume-averaged stress and strain, respectively. σ and ε denote the element stress and strain, respectively. V denotes the volume of RVEs. For RVEs in 2D, their resulting mechanical behavior is represented as a second-order homogenized elastic modulus tensor $\hat{C}_{ij} = [C_{11}, C_{12}, C_{13}; C_{21}, C_{22}, C_{23}; C_{31}, C_{32}, C_{33}]$. The modulus tensor can be obtained by applying volume-averaged strain $\overline{\varepsilon}_{ij}$ on the RVEs, computing the volume-averaged stress $\overline{\sigma}_{ij}$ by finite element simulations, and solving the stress-strain relationship defined as $\overline{\sigma}_{ij} = \hat{C}_{ij}\overline{\varepsilon}_{ij}$.

2.2.2 RVE homogenization method

Recently, Sadik [16] developed an ABAQUS CAE plugin called EasyPBC to determine the homogenized mechanical properties of user-defined RVEs. In addition to the computational efficiency and convenience of the commercial finite element software (ABAQUS), this plugin can model the composite materials, and the materials with void inclusions hence holding the potential for broader applications in materials design [16]. The concept of this plugin is similar to the asymptotic homogenization method. The effective properties of RVEs can be achieved by imposing unit strain tests on micro-scale RVEs (Fig. 2.4). In the post-processing of the ABAQUS, the effective properties of the RVE are determined by the equations, illustrated as:

$$E_{ij} = \frac{\text{Axial Tensile}}{\text{Axial strain}} \tag{2.8}$$

$$\nu_{ij} = -\frac{\text{Transverse strain}}{\text{Axial strain}} \tag{2.9}$$

$$G_{ij} = \frac{\text{Shear stress}}{\text{Product of applied shear strain}} \tag{2.10}$$

2.2.3 Boundary conditions

The choice of boundary conditions depends on distributed patterns of RVEs within the design domain. Specifically, it relies on whether the RVEs are periodically distributed within the design domain. This study employed two types of boundary conditions, including the periodic boundary condition (e.g. porous metamaterials) and the affine displacement boundary condition (e.g. spinodoid metamaterials), which are defined by Eq. 2.11 and Eq. 2.12, respectively [52].

$$u_i(x_j^+) - u_i(x_j^-) = \overline{\varepsilon}_{ij},\ t_i(x_j^+) + t_i(x_j^-) = 0 \tag{2.11}$$

$$u_i = \overline{\varepsilon}_{ij} x_j \tag{2.12}$$

where u denotes the nodal displacement on the surface of RVEs, $\overline{\varepsilon}$ denotes the volume-averaged strain over an RVE, and t denotes the surface traction.

2.3 Optimization Methods

The study employs two optimization methods to address the design problems. The first method is called sequential quadratic programming (SQP), which is a gradient-based method. It is applied to solve topology optimization problems with extreme properties in the study (Ch. 7). Another method is a gradient-free optimizer called genetic algorithm (GA). Due to its stochastic search algorithm based on the principle of natural evolution, the study utilizes the GA not only to explore complex parameter spaces of CMMs (Ch. 3) but also to sample property points in a material property chart (Ch. 4). The mathematical formulation of two optimization methods is illustrated next.

2.3.1 Sequential quadratic programming

Sequential quadratic programming (SQP) is a gradient-based optimization method for constrained nonlinear problems. Given the design variables x and all corresponding objective functions (f) and active constraints (A), this algorithm solves a sequence of optimization

sub-problems to determine the search direction (d). The sub-problem is defined as [39]:

$$\begin{aligned} &\text{Minimize} : f(x) = \tfrac{1}{2} d^T \bigtriangledown^2 f^{(k)} d + \bigtriangledown^T f^k d \\ &\text{Find} : d \\ &\text{Subject to} : Ad = 0 \end{aligned} \tag{2.13}$$

The search direction of design variables can be determined from the solution of the objective (Eq. 2.13) subjected to the linearization of the constraints that are expressed as [39]:

$$\begin{bmatrix} \bigtriangledown^2 f^{(k)} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} - \bigtriangledown f^{(k)} \\ 0 \end{bmatrix} \tag{2.14}$$

where λ is Lagrange multipliers of the Karush–Kuhn–Tucker (KKT) conditions. After solving its sub-problem, the SQP updates design variables as [39]:

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)} \tag{2.15}$$

where the step size $\alpha^{(k)}$ is determined by the line search method.

2.3.2 Genetic algorithm

Genetic algorithm (GA) is a stochastic global search method inspired by Charles Darwin's theory of natural evolution. This algorithm undergoes the process of nature selection during the optimization process. In each generation of the GA, the algorithm selects the fittest individuals within the current population by applying K-Way tournament selection [54]. Then, the new offspring of the next generation are produced by uniform crossover with a prescribed crossover rate. With the uniform crossover, each geometric parameter of repro-

duced individuals is randomly selected from one of the corresponding parameters of the parent chromosomes [55]. Meanwhile, the uniform mutation increases the diversity of the individuals and enables the GA to perform a more extensive exploration of parameter space. With the uniform mutation in the GA, each geometric parameter is randomly selected for mutation with an associated probability rate [56]. This probability rate is called the mutation rate and determines the probability of the selected parameters being mutated. The algorithm is terminated when either the maximum iteration number or the stall generation limit is reached. The flowchart of the genetic algorithm is shown in Fig. 2.5.

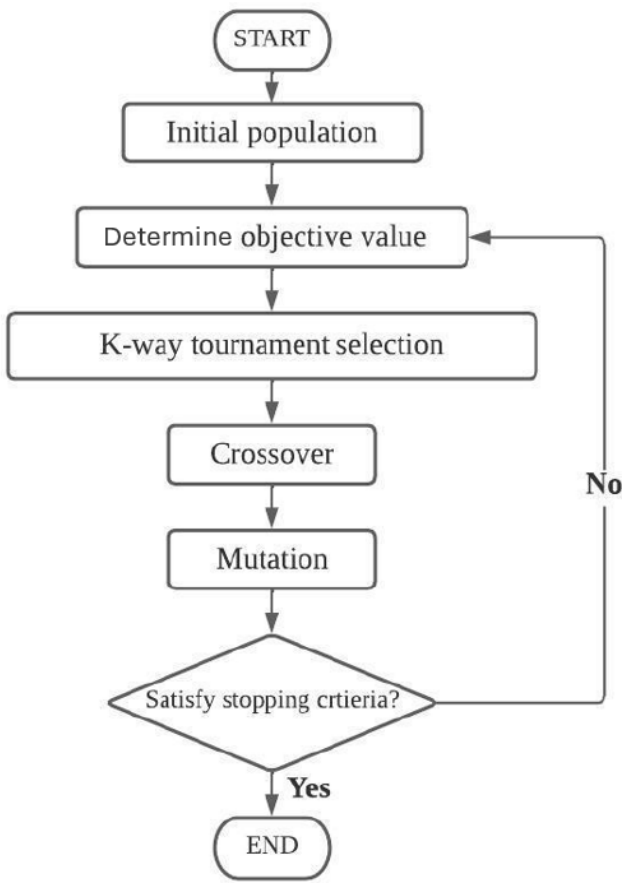


Figure 2.5: Flowchart of the genetic algorithm.

Besides, following the works of Maaranen et al. [13, 14], various random point processes can be studied to understand the impact of the initial population on the final objective function

values over several generations of the GA. The two inhibition point processes studied in the works of Maaranen et al. [13, 14], simple sequential inhibition process and nonaligned systemic sampling, have a disadvantage in either computational seed or usability. A simple sequential inhibition process requires a number of trial points to satisfy a pre-defined distance metric. Besides, nonaligned systemic sampling does not allow users to freely control the sampling size, which is determined by the grid size and dimension of the parameter space. Therefore, with the consideration of their efficiency and usability, these two inhibition processes are not studied in this work. Latin hypercube sampling (LHS) is a simple but efficient approach to distribute the sample points evenly across the parameter space [57]. It can be an alternative random point process to initialize the GA population. Another alternative point generator can be a quasi-random sequence generator, such as Sobol and Niederreither sequence generators. Their generated samples have a low discrepancy property and a relatively uniform distribution in the unit hypercube compared to a pseudo-random number generator. Figure 2.6 illustrates examples of 2D initial populations with 16 individuals generated using the four random point processes.

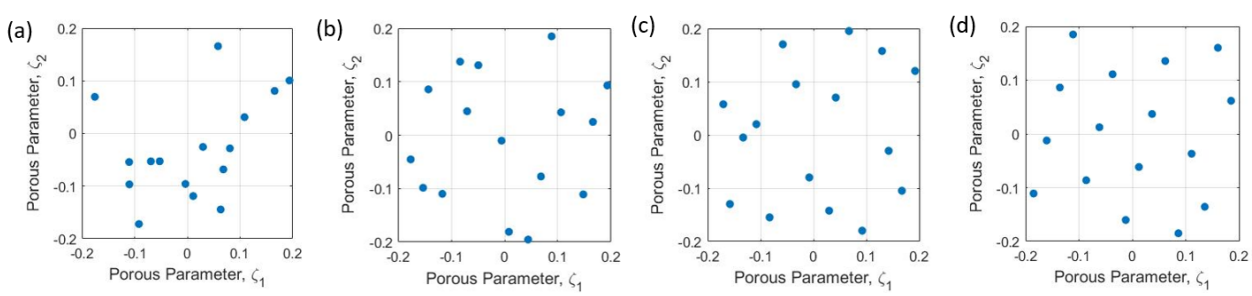


Figure 2.6: Sample 2D initial populations generated by (a) pseudo-random number generator, (b) Sobol sequence generator, (c) Niederreither sequence generator, and (d) Latin hypercube sampling (LHS).

2.4 Supervised Machine Learning Models

The application of ML algorithms aims to build structure-property relations of metamaterials. With an NN-based surrogate model, Monte-Carlo sampling for uncertainty quantification can be accelerated by bypassing the FE analysis with high computational expense. The CNN-based surrogate model provides a tool to explore such relations for metamaterials with microstructures that may not be characterized by geometric variables. Instead, the CNN model has the capability to extract geometric features directly from the input images to predict their corresponding properties. Furthermore, recent advances in deep learning techniques (e.g., CGAN) allow us to generate new images of microstructures with a given type. The generated images have similar or equivalent features as the images from the input database. Consequently, these ML models can benefit the research on the development and discovery of materials.

2.4.1 Multiple layer perception

In deep learning, a neural network (NN) is built as a connected multi-layer neural network called a Multi-layer Perceptron (MLP) shown in Fig 2.7. Its hidden layers are made up of a set of neurons with parameter sets (ω) containing the weights and biases. MLP is applied to materials whose geometric patterns of RVEs can be parameterized. With inputs of the geometric parameters, MLP bypasses the expensive FE simulations and predicts their corresponding structural properties efficiently. The objective of training the surrogate model of NN is to minimize the loss between actual values and predictions concerning the NN parameters, ω. The performance of NN is evaluated in terms of the mean squared error

(MSE), and it is defined in Eq. 2.16 [58]:

$$\min_{\omega} MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.16}$$

where y_i is the actual property of CMMs, and $\hat{y}_i$ is the property predicted by the trained surrogate model.

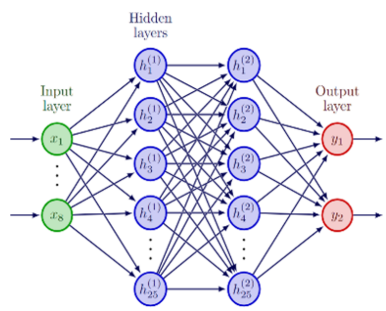


Figure 2.7: Architecture of MLP.

2.4.2 Convolutional neural network

It is not possible to parameterize the geometric variables of all microstructures. Some 2D microstructures can only be quantified using an image-based approach represented in pixel levels. The convolutional neural network (CNN) can take in an input image of these microstructures and efficiently predict their corresponding properties in a forward direction. Commonly, the CNN is formed into two sections, the feature maps and the fully connected layers. The first part contains 2D convolution layers (CON2D). They extract features from input images. Convolution layers perform the downsampling operation and are followed by max-pooling layers, which help to identify the most distinctive features of downsampled images [59]. The outputs of the feature map are flattened and directly connected with fully connected layers (Dense) to predict the numerical values of properties by using regression analysis [60]. Apart from the input design variables (MLP: geometric variables; CNN: pixel

values), the loss function is defined as the same equation of MLP (Eq. 2.16). Fig. 2.8 illustrates a CNN for predicting the effective properties of spinodoid metamaterials with inputs binary images of their microstructures.

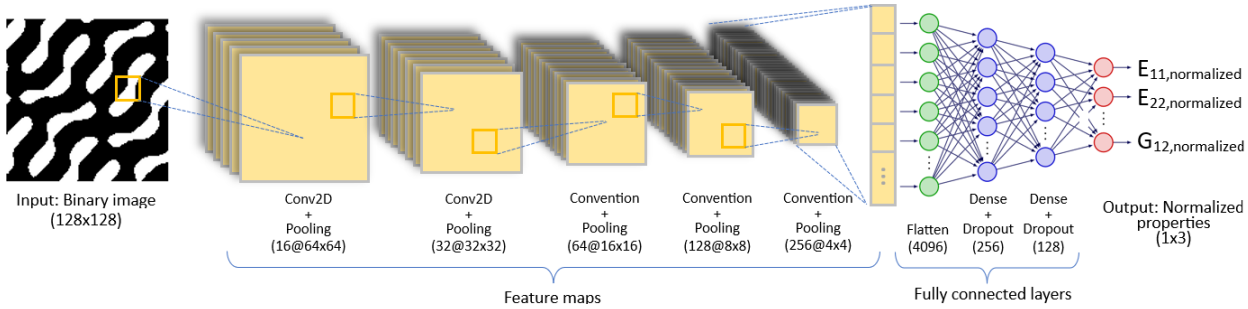


Figure 2.8: Architecture of CNN for predicting properties of spinodoid metamaterials in a forward direction.

2.4.3 Conditional generative adversarial network

The conditional generative adversarial network (CGAN) is a type of deep-learning model that achieves data generation by incorporating labels as constraints [61]. Its generated data has equivalent statistics as the training set. Compared to the classic generative adversarial network (GAN), the CGAN allows the users to generate images of a given type [62]. The architecture of CGAN is comprised of two sub-networks: a generator (G) and a discriminator (D). The CGAN is used in the context of materials design in this work. In this materials design application, the generator takes a random noise vector from the latent space and a target property label as inputs. During the adversarial process, it learns to generate new fake geometric patterns indistinguishable from the training data for specific target properties [63]. Meanwhile, the discriminator learns to distinguish if its input of geometric patterns is real (from the training data) or fake (produced by the generator) [63]. The two networks are trained concurrently. As the discriminator identifies fake images more precisely, the generator has to generate more realistic data to fool the discriminator.

However, the conventional CGAN might suffer from the overfitting problem as training the discriminator is excessively dependent on the training data [64]. To overcome this problem, an independent solver (S) promotes the generator to yield geometric patterns with objective (target) properties. The solver is a convolutional neural network (CNN) applied to bridge the geometric patterns and properties (forward problem). The solver is applied to promote the generator to create the arrangement with target properties. In the meanwhile, the assessment process of the discriminator pushes the generator to produce realistic images of the microstructures. With the same input label, a well-trained generator could rapidly provide a group of candidate solutions with exact or similar target properties by modifying an input of random noise vectors from the latent space. The loss function of the generator and the discriminator are defined in Eq. 2.17 and Eq. 2.18, respectively. Fig. 2.9 illustrates the general architecture of the CGAN facilitated by the solver.

$$L_G(\omega^{(G)}) = \frac{1}{N}\sum_{i=1}^{N}\{E_z[log(1 - D(G(z|y_i))] + \alpha(y_i - \hat{y}_i)^2\} \tag{2.17}$$

$$L_D(\omega^{(D)}) = -\frac{1}{N}\sum_{i=1}^{N}\{E_x[logD(x|y_i)] + E_z[log(1 - D(G(z|y_i))]\} \tag{2.18}$$

where $\omega^{(G)}$ and $\omega^{(D)}$ are the parameters of the generator and discriminator, respectively. z is the random noise vector, and α is a weighting coefficient that determines the extent to which the loss functions affect the generator. x shows the real images from the database. The architectures of G and D are shown in Fig. 6.6 and 6.7, respectively.

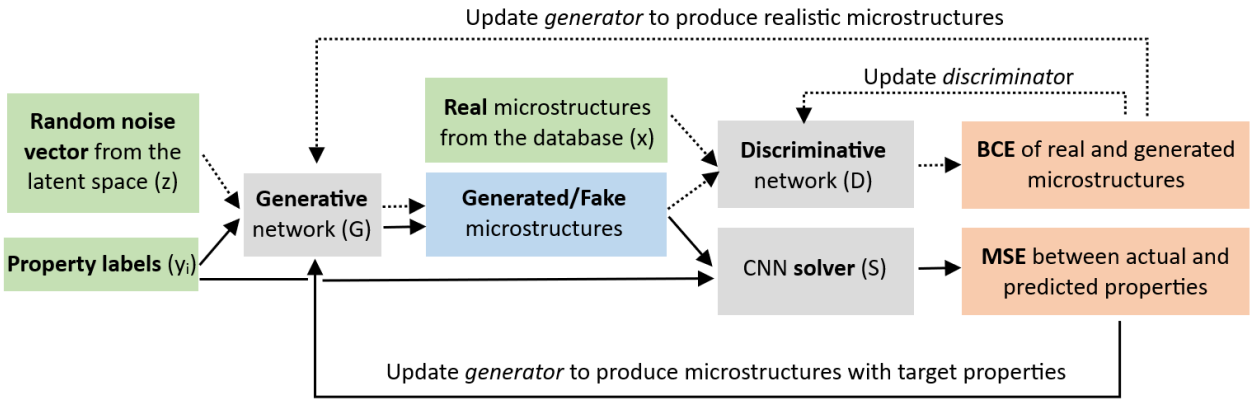


Figure 2.9: Architecture of the conditional generative adversarial network facilitated by the solver (BCE denotes Binary Cross Entropy, which is a loss function to measure the difference between actual binary labels from the database and predicted binary outcomes from the discriminator).

Chapter 3

PARAMETER SPACE EXPLORATION OF POROUS METAMATERIALS

Chapter 3 outlines an experience-free and systematic framework using a multi-objective genetic algorithm (GA) with a small population size and conditional search strategy for the inverse design of porous metamaterials. For the inverse design, the optimization problem is formulated in Sec. 3.1. Section 3.2 introduces the proposed inverse design framework with micro-GA, the alternative point generators for generating the initial population of GA, and the method for evaluating uniformity of point distributions. The optimization results of porous metamaterials with PUCs and non-PUCs are summarized in Sec. 3.3. The section also demonstrates the impacts of an initial population on optimal solutions obtained by GA. In Sec. 3.4, this study compares the micro-GA with the standard GA in aspects of their efficiency and accuracy, and discusses the potential issues of deep learning methods and the advantages of our proposed design framework.

3.1 Optimization Formulation

In this study, the inverse design of micro-structured materials can be formulated as a multi-objective optimization problem to minimize the difference between the objective and predicted effective properties of the RVEs. This study primarily focuses on two mechanical properties of porous metamaterials, effective Young's modulus and effective Poisson's ratio, which are obtained by the RVE homogenization method in Sec. 2.2.2. The mathematical formulation of the optimization problem reads as follows:

$$\begin{aligned}
&\text{Minimize}: f(\zeta) = [\tilde{w}_1(\nu_{12} - \nu_0)]^2 + [\tilde{w}_2(E_{11} - E_0)]^2 \\
&\text{Find}: \zeta = ([\zeta_1, \zeta_2], ..., [\zeta_{n-1}, \zeta_n]) \\
&\text{Subject to}: \zeta^L \leq \zeta_n \leq \zeta^U (n = 1, 2, ..., 8) \\
&\qquad\qquad L_{tolr} \leq \tfrac{L}{2} - rsin(\theta), L_{tolr} \leq \tfrac{L}{2} - rcos(\theta)
\end{aligned} \tag{3.1}$$

where ζ denotes the porous parameters, $\tilde{w}_i$ is a weighted coefficient to scale the two effective properties in the same numerical level, and n is the number of porous parameters which is equal to twice as many unit cells within the RVE (see Fig. 2.1), ζ^L and ζ^U are the lower and upper bounds of porous parameters which are equal to -0.2 and 0.2, respectively, and f is the fitness function of the Micro-GA which is the summation of absolute errors between the target (v_0, E_0) and the predicted properties (v_{12}, E_{11}). The geometric constraint ensures the continuity of structures. It eliminates the convergence issues in the FE simulations when the distance from the edge of the pore shape to the boundary of the unit cell is minimal. Within $0 \leq \theta \leq 2\pi$, the minimum distance should be larger than L_{tolr} as shown in Fig. 3.1.

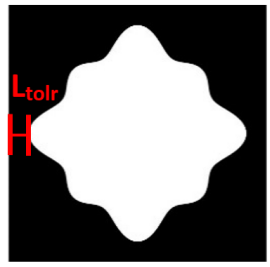


Figure 3.1: Constraints of the porous shapes in RVEs.

3.2 Methodology for Micro-Genetic Algorithm

Here, we propose an experience-free and heuristic searching strategy based on a micro-genetic algorithm (μGA) to overcome the issue of gradient-based methods and traditional genetic algorithms. Compared with the deep learning method, our method does not require any prior information on the data domain and can discover the complex non-linear relationships between the geometries of RVEs and their corresponding properties by evolutionary algorithms. Considering the computational expense of the standard genetic algorithm (SGA), we employ μGA, which undergoes SGA operators including tournament selection, crossover, uniform mutation, and elitism but requires a small population size in each cycle. Obviously, with a small population size, SGA has difficulty maintaining diversity over several generations. Moreover, it could induce premature convergence due to the presence of a local optimum solution. To avoid this problem, the basic idea of μGA is to restart SGA with a small population size (N_P) in multiple cycles. Once the diversity of the current cycle is low, the fittest individual is kept as a reference individual, and N_P-1 new individuals from the initial population are introduced to the next cycle of the SGA. Meanwhile, the range of the parameter space remains the same. The diversity of the generations in the μGA is evaluated by the overall change of fitness values among the stall generation limits. The basic

idea allows μGA to continuously search for global optimum solutions and avoid the current searching step trapping in local minima of our optimization problem.

In μGA, the searching strategy continuously processes a new cycle of SGA while introducing the new fittest individual until the maximum iteration number or error tolerance is reached, where the error tolerance is the maximum relative error between the objective and optimal properties. The fittest individual in the early stages of the searching process may already have outstanding values of the objective (fitness) function, but may not satisfy the error tolerance of the stopping criteria. In the successive few cycles of SGA, the iterated searching process might be unable to improve the local optimum solution further. With the restart of SGA sufficient times, the iterated search process has a possibility to optimize the current fittest individual approaching the objective value. However, for a large number of restarts in SGA, the iterated search process is time-consuming, even with the small population size. Here, we assume that the current fittest individuals (local minima), achieved in the case of no improvement in the optimum solution over several cycles, have a significant possibility of being located near the global minima. The assumption was proved by the experiments with RVEs containing one periodic unit cell (Fig. 2.1a). In the parameter space, we evaluated the Euclidean distances between 50 local minima, determined by 50 runs of the SGA, and global minima, defined by a known geometric pattern of an RVE with corresponding homogenized properties. Table 3.1 shows ten identical global minima with the average distance of their 50 local minima. In Table 3.1, the average distances for all the cases are less than 50% of the width of the parameter space 0.4, which is determined based on the pre-defined range of porous parameters from -0.2 to 0.2. The average distance in case 4 even reaches 4% of the width. Inspired by the results, we can reduce the parameter space based on the current fittest individuals of the SGA to decrease the computational cost. Considering a more complex design scenario, like the one with a higher dimensional parameters space and

a Pareto optimal set, a new parameter space cannot simply be reduced based on a single local minimum and a specifically designated width. The SGA with a new reduced space should be determined by referring to at least two local minima, which allows us to obtain new fittest individuals with extremely high accuracy that satisfies the error tolerance to stop the searching process. The statement is validated by the results in Section 3.4.

Table 3.1: Summary of Euclidean distances between local minima and one global minimum.

Example index	Global minimum	Corresponding properties, $[\nu_0, E_0]$	Average Euclidean distance of 50 local minima
1	[0.1800 -0.1367]	[0.1843 1111.87MPa]	0.1393
2	[-0.1435 0.0048]	[0.1939 1417.06MPa]	0.0527
3	[0.0885 0.1715]	[0.1571 994.75MPa]	0.0772
4	[0.0928 0.0999]	[0.1853 1137.40MPa]	0.0765
5	[-0.0371 -0.1042]	[0.2020 1289.35MPa]	0.0481
6	[0.1370 0.0652]	[0.1896 1107.50MPa]	0.1165
7	[0.1011 -0.0013]	[0.2095 1274.87MPa]	0.0248
8	[0.0188 -0.0388]	[0.2152 1372.35MPa]	0.0174
9	[-0.1572 0.0897]	[0.1680 1357.84MPa]	0.0278
10	[-0.1060 -0.0404]	[0.2049 1393.67MPa]	0.0288

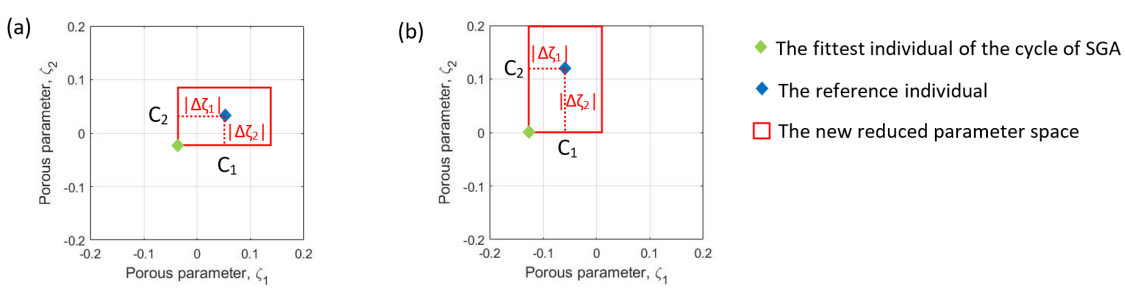


Figure 3.2: Construction of the new reduced parameter space.

Therefore, the new range of the n-dimensional parameter space is referred by the current fittest individual (the first local minimum) with $[\zeta_{1,0}, \zeta_{2,0}, ...\zeta_{n,0}]$ and the new fittest individual (the second local minimum) with $[\zeta_{1,1}, \zeta_{2,1}, ...\zeta_{n,1}]$, achieving in the new run of SGA without any reference individual in the initial population. The width of the new n-dimensional

parameter space is determined as twice as the absolute difference of the porous parameters in each dimension, $2 \times |\zeta_{n,0} - \zeta_{n,1}|$, as demonstrated in Fig. 3.2. Once the side lengths of reduced parameter space (C_i) in each dimension are determined, the new cycle of SGA is processed without any reference individual in the initial population. The fittest individual in the new cycle can be continuously optimized by following the basic idea of μGA in the reduced parameter space. The following pseudo-code illustrates the detailed procedure of our global searching strategy:

function Micro-GA

begin

Define porous parameters $\zeta \in [\zeta_{0,min}, \zeta_{0,max}]$ and number of cycles of SGA(i) = 0

<u>case 1:</u> (1). Initialize the population (P) with size N_P

(2). Optimize the initial population by SGA

(3). Save the fittest individual (K_{best}) and its objective function value (f_{best})

(4). i = i + 1

while f_{best} > tolerance ***or*** i $\leqslant$ maximum iteration ***then***

<u>case 2:</u> (1). Introduce K_{best} and initialize P with N_P-1

(2). Optimize the initial population by SGA

(3). Save the fittest individual (K_{best}) and its objective function value (f_{best})

(4). i = i + 1

if $f_{best} \leqslant$ tolerance ***or*** i > maximum iteration ***then***

end function

else

repeat

begin

Run ***case 2***

i = i + 1

if $f_{best} \leqslant$ tolerance ***or*** i > maximum iteration ***then***

end function

end

end

until f_{best} of the current cycle of SGA = f_{best} of the previous cycle of SGA

Save f_{best} of the current cycle ($f_{0,best}$) with $[\zeta_{1,0}, \zeta_{2,0}, ...\zeta_{N_P,0}]$

Run ***case 1***

Save f_{best} of the new cycle of SGA ($f_{0,best}$) with $[\zeta_{1,1}, \zeta_{2,1}, ...\zeta_{N_P,1}]$

case 3: (1). Define the new range of ζ_{N_p} in N_P dimensions

begin

while j $\leqslant N_P$

Determine absolute difference between $\zeta_{j,0}$ and $\zeta_{j,1}$ in j^{th} dimension,

$\triangle\zeta_j = |\zeta_{j,0} - \zeta_{j,1}|$

if $\zeta_{j,1} + \triangle\zeta_j \geqslant \zeta_{0,max}$

$\zeta_{j,max} = \zeta_{0,max}$

else

$\zeta_{j,max} = \zeta_{j,1} + \triangle\zeta_j$

end

if $\zeta_{j,1} - \triangle\zeta_j \leqslant \zeta_{0,min}$

$\zeta_{j,min} = \zeta_{0,min}$

else

$\zeta_{j,min} = \zeta_{j,1} - \triangle\zeta_j$

end

end

(2). Run ***case 1*** with a new parameter space defined by $\zeta_{j,max}$ and $\zeta_{j,min}$

(3). Save the fittest individual (K_{best}) and its objective function value (f_{best})

(4). i = i + 1

(5). ***if*** $f_{best} \leqslant$ tolerance ***or*** i > maximum iteration ***then***

end function

else

repeat

begin

Run ***case 2***

i = i + 1

end

until $f_{best} \leqslant$ tolerance ***or*** i > maximum iteration

end

end function

3.3 Results

3.3.1 Preliminary settings

In this section, the effectiveness of the inverse design method is demonstrated by testing two numerical examples: RVEs with periodic unit cells and RVEs with non-periodic unit cells. The parameterization describes the unit cells of RVEs in Eq. 2.1 and 2.2. In the first numerical example (2D study case), the geometric patterns of RVEs consist of 1×1 periodical unit cells, which is defined by 1×2 geometric vector $[\zeta_1, \zeta_2]$ shown in Fig. 2.1a. The second numerical example (8D study case) has RVEs consisting of 2×2 non-periodic

unit cells, whose geometric patterns are defined by a 2×4 geometric matrix, $[[\zeta_1, \zeta_2]\ [\zeta_3, \zeta_4];\ [\zeta_5, \zeta_6]\ [\zeta_7, \zeta_8]]$ shown in Fig. 2.1b. The chosen sizes of RVEs in the two numerical examples are large enough to represent the microstructure without introducing non-existing properties [65], and small enough to reduce the computational expense in FE analysis. In our study, the RVEs in the two numerical examples are assigned to an isotopic and linear elastic material called unfilled Polyetherimide (Young's modulus = 3580 MPa, Poisson's ratio = 0.35 [67]). The range of porous parameters ζ is identical in each dimension, from -0.2 to 0.2. The size of the unit cell is defined by a unit side length, $L = 1$. The smallest distance L_{tolr} between the edge of the pore shape and the boundary of the unit cell is designated as 0.05. The porosity of unit cells ϕ is equal to 0.4. Besides, the optimization problem in this section is solved using the μGA with the same mutation rate = 0.1, crossover rate = 0.8, population size = 16, and stall generation limit = 32. In the inner loops of our searching methodology, all cycles of SGA have the same stopping criteria: (i) the change of the optimal value of the fitness function during restarting μGA should be less than 1e-4, and (ii) the maximum iteration number should be smaller than 50. In addition, there are two stopping criteria for the outer loops of our searching methodology: (i) the maximum number of cycles should be less than 10, and (ii) the final optimal properties should satisfy a certain relative error between the objective and optimal properties. The two objective properties for this study are chosen to be the effective Young's modulus and Poisson's ratio of the CMMs. In addition, we intend to select combinations of objective properties that cover the design spaces of the 2D and 8D study cases, and meanwhile ensure the diversity of expected geometric patterns of RVEs, which have various curvatures of the porous surfaces in unit cells. The following numerical examples demonstrate that the presented approach successfully identifies all these different geometries while optimizing the geometric patterns with preferred combinations of the two effective properties.

3.3.2 Optimization of RVEs with periodic unit cells

Suppose a geometric pattern of the 1×1 RVE with the periodic unit cell is desired that exhibits objective effective Young's modulus and Poisson's ratio. It corresponds to a 2D study case with 2 design parameters $[\zeta_1, \zeta_2]$. In the 2D study case, the tolerance of the relative error with respect to the objective properties is defined as 0.05% in comparison to the objective properties. In the beginning of our optimization framework, there is no *prior* information about the local or global minima available for initializing the population of SGA. Therefore, uniformly distributed sample points with diverse values and locations might be useful for searching optimum solutions throughout the 2D parameter space [14]. Traditionally, the initial population of SGA is generated by pseudo-random number generator (PRNG) [13]. Our study compares its effect on objective function values to three alternative random point generators, including the Sobol sequence generator (SSG), Niederreither sequence generator (NSG), and Latin hypercube sampling (LHS). The empirical empty space statistics (ESS) function evaluates their goodness for uniform distribution property, which is defined as [13]:

$$ESS(r) = 1 - Pr(B(x, r) \text{ is empty}), \tag{3.2}$$

where x is a randomly chosen point in the feasible region, $B(x,r)$ is a ball centered at x with radius r, and Pr denotes the probability [13]. The function randomly assigns the ball B with the radius r at the center x in the feasible region. Then, the probability of the function is the likelihood of having no sample points in the ball B. Figure 3.3a illustrates the relationship between the radius of the ball B and the average *ess* function values of hundreds of point generations. As the radius increases, the ball B is less likely to be empty, leading to an increase in the *ess* function. If the emptiness of the ball B is more sensitive to the change of radius and less depends on its center x, the curve tends to be steep. In other words,

the generator with more evenly distributed sample points within the parameter space has a steeper curve of the *ess* function. Hence, the sequence from the highest to the lowest in the uniformity of sample distribution is LHS, NSG, SSG, and PRNG, respectively. In Fig. 3.3a, LHS first converges to the maximum value of the *ess* function, representing that it has the highest uniformity among the four generators.

In addition to the uniformity, we check the accuracy of the four random point generators. The accuracy is evaluated by solving the optimization problem with 20 different objective properties shown in Table 3.2. Due to the minimization problem, the smaller the objective function value is, the higher the accuracy of the optimum solution is. We solve each problem with four variants of the SGA a hundred times. The benchmark results in Fig. 3.3b are the objective function values obtained with PRNG. The height of the bars in Fig. 3.3b represents the decrease in the objective function values compared to the benchmark. Table 3.2 summarizes the results from Fig. 3.3b, including the average decrease in objective function values in percentage, the number of improved objective function values, and the variance of the decreases in objective function values in percentage of all tested objective properties. Generally, the objective function values can be successfully improved by replacing any of the three alternative generators with the benchmark. Although LHS reduces a large number of objective function values in percentage, its performance or stability varies considerably from case to case by referring to the relatively large values of the variance in Table 3.2. The overall results from Fig. 3.3b and Table 3.2 prove that the distributions of the initial population have an important effect on the objective function values in the 2D case. However, the comparison of the variance in Table 3.2 indicates that the highest uniformity of the initial population does not necessarily contribute to the best performance in optimizing the geometric patterns of RVEs. Overall, based on the comprehensive evaluations from Table 3.2, we choose either NSG or SSG as the random point generator of SGA in our search strategy.

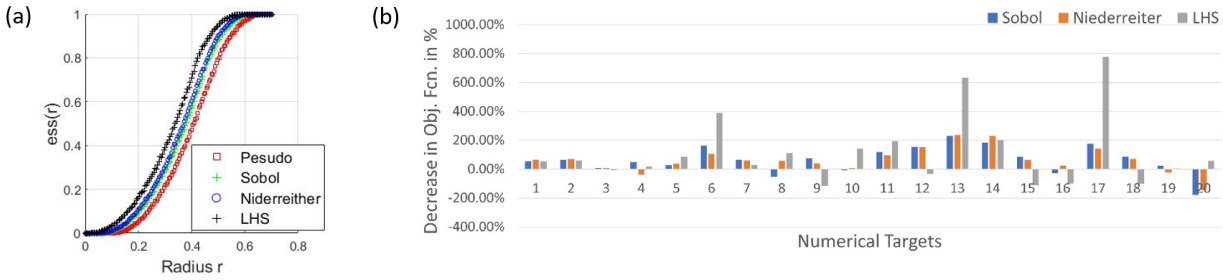


Figure 3.3: Comparison of the four random point generators in 2D parameter space: (a) *ess* function vs. radius and (b) decrease in objective function values.

Table 3.2: Summary of results from Fig. 3.3b (2D).

Random point generator	Average decrease in obj. fcn in %	Number of the improved obj. fcn	Variance in obj. fcn
SSG	65.32%	16	0.8878
NSG	63.55%	17	0.7621
LHS	114.97%	14	5.6341

After evaluating the four random point generators, we process our complete searching strategy with a quasi-random point generator, such as SSG, to solve the optimization problem with 20 different objective properties shown in Table 3.3. Table 3.3 numerically summarizes the optimization results, including the objective properties, the order of SGA cases from the pseudo-code in Sec 3.2, the relative errors in percentage compared to the objective properties, and the total number of generations in SGA. Owing to the benefits of the quasi-random point sampling, the objective properties in the several examples can be achieved only in a single cycle shown in Examples #5, #6, #16, and #19. Fig. 3.4 visually demonstrates the collective converging histories of all cycles of SGA within the whole process of μGA. The history shows the logarithm of objective function values over the generations (iteration steps). In Fig. 3.4, the solid vertical lines split the history into several sections, representing a cycle of SGA and denoted with the corresponding case index from the pseudo-code. Some of the example problems in Table 3.3, such as Examples #1, #2, #7, #12, and #17, can be

Table 3.3: Summary of the optimization results (2D).

Example index	Objective properties, $[\nu_0, E_0]$	Order of SGA cases	Relative error, $[\%\nu, \%E]$	Total number of generations
1	[0.1755, 1068.33MPa]	12	$[0.010\%, 0.037\%]$	84
2	[0.1968, 1442.13MPa]	122	$[0.045\%, 0.027\%]$	108
3	[0.2171, 1408.04MPa]	1213	$[0.002\%, 0.011\%]$	111
4	[0.2059, 1251.02MPa]	121	[0.020%.0.015%]	134
5	[0.2120, 1378.09MPa]	1	$[0.004\%, 0.001\%]$	32
6	[0.1932, 1401.69MPa]	1	$[0.033\%, 0.008\%]$	47
7	[0.2175, 1349.87MPa]	12	$[0.002\%, 0.012\%]$	78
8	[0.1917, 1240.80MPa]	12213	$[0.001\%, 0.000\%]$	186
9	[0.1838, 1069.64MPa]	121	$[0.010\%, 0.007\%]$	81
10	[0.2116, 1440.36MPa]	12132	$[0.002\%, 0.033\%]$	160
11	[0.1731, 1363.41MPa]	1221	$[0.000\%, 0.000\%]$	128
12	[0.2077, 1431.43MPa]	12	$[0.001\%, 0.000\%]$	38
13	[0.2077, 1326.66MPa]	12132	$[0.005\%, 0.050\%]$	177
14	[0.1591, 1273.32MPa]	12132	$[0.002\%, 0.000\%]$	185
15	[0.1845, 1110.27MPa]	121	$[0.002\%, 0.000\%]$	92
16	[0.1910, 1129.32MPa]	1	$[0.000\%, 0.000\%]$	35
17	[0.1857, 1180.21MPa]	122	$[0.037\%, 0.033\%]$	104
18	[0.1522, 1184.82MPa]	121	$[0.000\%, 0.000\%]$	87
19	[0.1731, 1189.14MPa]	1	$[0.000\%, 0.000\%]$	45
20	[0.1866, 1140.06MPa]	121	$[0.000\%, 0.000\%]$	86

directly solved by the basic idea of μGA combined with case #1 and #2 from the pseudo-code. Some of the example problems, such as Examples #4, #9, #11, #15, #18, and #20, require restarting the SGA without any reference individual after the several cycles. Some other example problems, such as Examples #3, #8, #10, #13, and #14, require reducing the initial parameter space to improve the fittest individual further, which is stuck at the local optimum using case #1, case #3, and case #2 if necessary. Thus, the results shown in Table 3.3 and Fig. 3.4 numerically and visually prove the effectiveness of our search strategy, respectively.

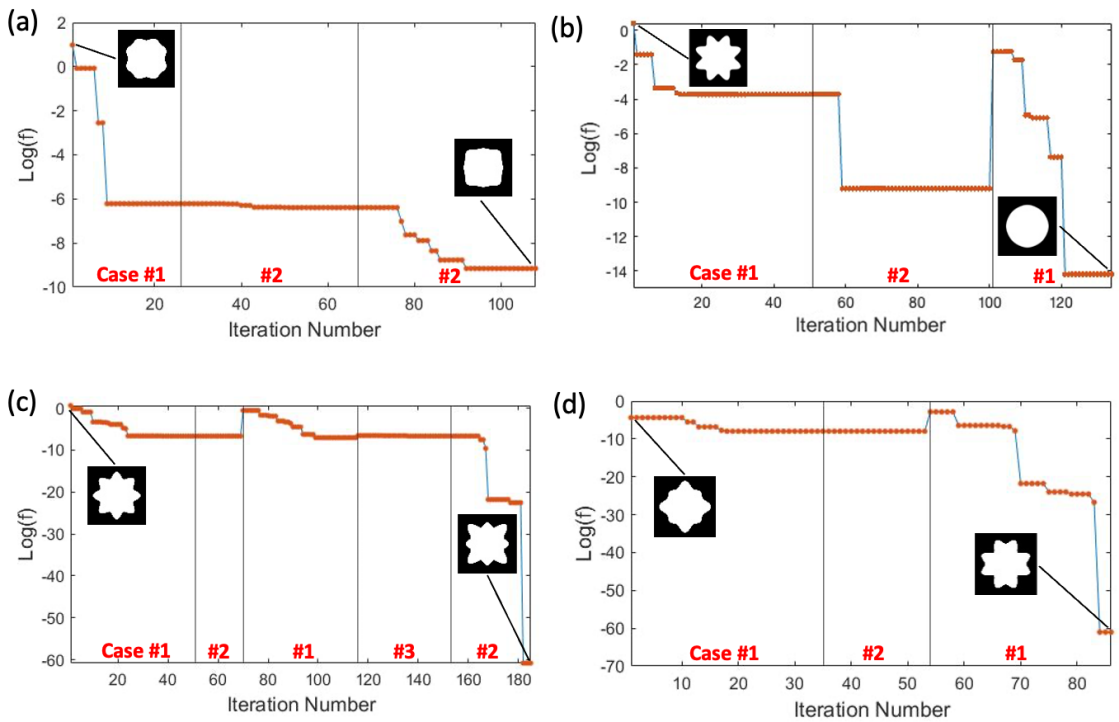


Figure 3.4: Combined converging history of four sample examples in Table 3.3: (a) example #2, (b) example #4, (c) example #14, and (d) example #20.

3.3.3 Optimization in RVEs with non-periodic unit cells

Suppose a geometric pattern of the 2×2 RVE with the non-periodic unit cell is desired that exhibits objective effective Young's modulus and Poisson's ratio. It is a 2D study case with 8 design parameters $[\zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5, \zeta_6, \zeta_7, \zeta_8]$. Since the higher complexity results in more significant variations in material properties, we set the tolerance of the relative error in the 8D study case as 0.01% to increase the reliability of our optimum results. Before processing our searching strategy within the optimization problem, we first evaluate the uniformity of the four random point generators by the *ess* function in Fig 3.5a. The population size of SGA is assigned as 16, which is the same as the 2D study case. In Fig. 3.5a, we found that the converging behaviors of the four random point generators are very similar, and their converging history plots are almost overlapping. These results are caused

by the small population size chosen in the high-dimensional parameter space. Consequently, the changes in the random point generators in SGA slightly affect the initial population's point distribution; thus, their uniformity is nearly identical. Moreover, by processing a single cycle of SGA with 20 different objective properties, we determine whether the sample distribution affects the accuracy of the objective function values in the high-dimensional study case. Table 3.4 summarizes the results from Fig. 3.5b and shows that the four random point generators have similar performances in optimizing the geometric patterns of RVEs. Therefore, we conclude that, in our case scenario, changing the random point generators hardly influences the objective function values of the SGA with a small population size and a large number of design parameters. Therefore, we could choose any of the four random point generators to process our search strategy in this study case.

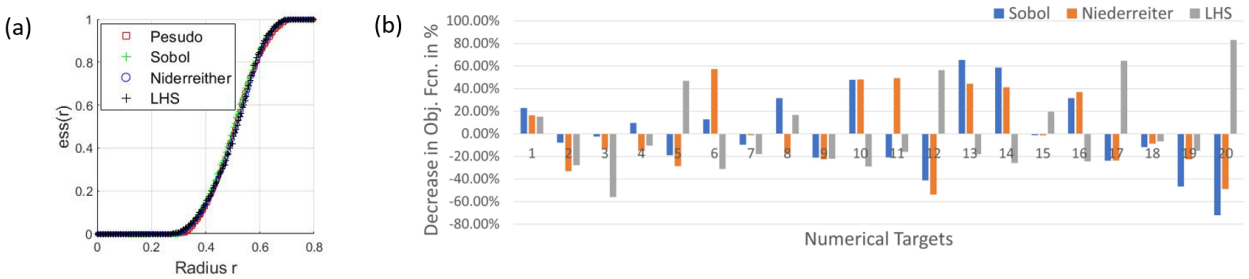


Figure 3.5: Comparison of the four random point generators in 8D parameter space: (a) *ess* function vs. radius and (b) decrease in objective function values.

Table 3.4: Summary of results from Fig. 3.5b (8D).

Random point generator	Average decrease in obj. fcn in %	Number of the improved obj. fcn	Variance in obj. fcn
SSG	0.16%	8	0.1261
NSG	0.20%	7	0.1546
LHS	0.18%	7	0.1372

The next step is to perform our complete searching strategy to solve the optimization with 20 different objective properties shown in Table 3.4. Like Table 3.2, the optimization results

are summarized in Table 3.4, including the objective properties, the order of SGA cases from the pseudo-code in Sec 3.2, the percentage errors compared to the objective properties, and the total number of generations in SGA. The growth of the complexities of the geometric patterns of RVEs provides sufficient freedom to design microstructures of the CMMs. It results in the possibility that RVEs with different geometric patterns share nearly identical effective properties. With a limited amount of candidate solutions, it is not difficult for the SGA in an 8D study case to achieve optimal solutions with a small error. This is shown by the results of Examples #4, #7, #8, #11, #19, and #20. Even without the updated method to generate the initial population, the increased freedom for material design allows these examples to achieve satisfactory results within a single cycle. Some of the example problems in Table 3.5, such as Examples #1, #10, #14, #15, #16, and #18, can continuously reduce the objective function values simply through the basic idea of μGA combined with cases #1 and #2 from the pseudo-code. Some of the example problems, such as Examples #5, #9, and #13, require dealing with premature convergence by neglecting the results from previous cycles instead of starting a new SGA without any reference individual. Some other example problems, such as Examples #2, #3, #6, #12, and #17, have the fittest individual, which is stuck at the local optimum over several cycles of SGA. Reducing the initial parameter space allows the problem to improve the individual further using cases #1, case #3, and case #2 if necessary. Moreover, the collective converging histories in Fig. 3.6 demonstrate the

optimization process over several cycles of SGA in a straightforward way. The histories clearly illustrate the issue of premature convergence, the restarts of SGA for a new fitness individual, and the continuous decrease in the objective function values. Overall, both numerical and visual results in the 8D study case indicate the effectiveness of our search strategy.

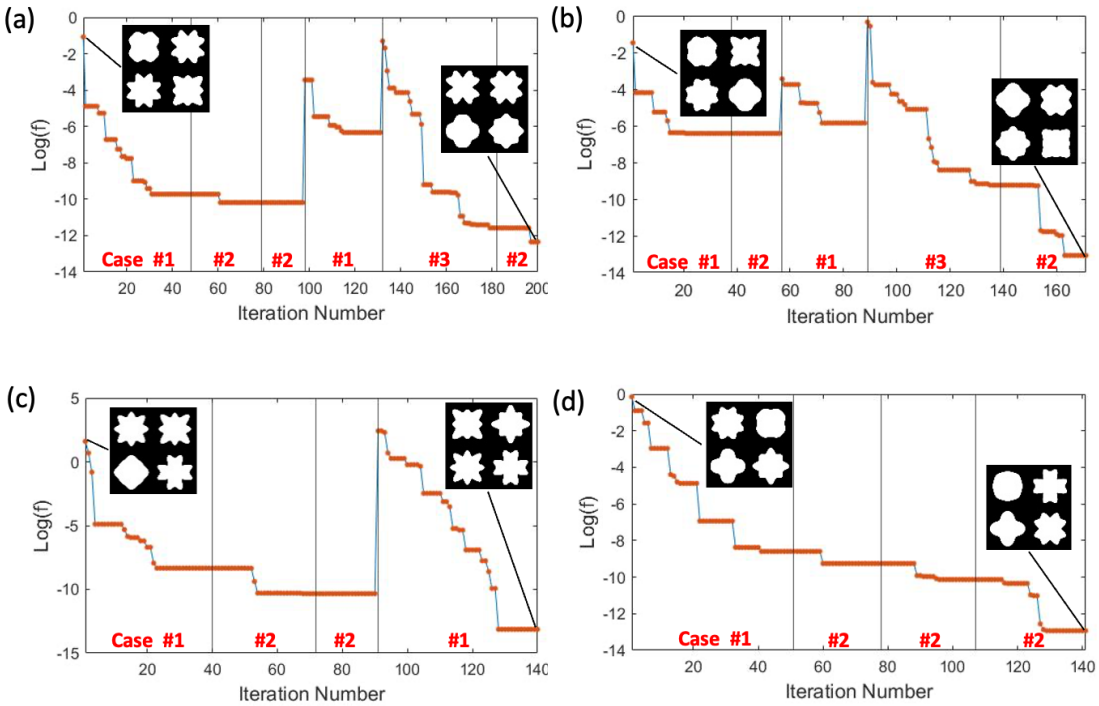


Figure 3.6: Combined converging history of four sample examples in Table 3.5: (a) example #3, (b) example #5, (c) example #9, and (d) example #15.

3.4 Discussion

3.4.1 Comparison of the results of standard and micro-GA

The results from the numerical examples demonstrate the effectiveness and reliability of our proposed inverse design method. In this section, we further illustrate the motivations for employing the micro-genetic algorithm (μGA) with the new searching strategy instead of the traditional/standard genetic algorithm (SGA). To make comparisons of two optimization methods, we set up one control group with the μGA and two experimental groups with the SGA. All test runs were performed on a laptop computer with a 2.60Hz Intel Core i7-10750H processor and 16GB DDR4 memory at 2133 MHz. All groups intend to solve the same inverse design problem with 20 objective properties from the previous section for 2D

Table 3.5: Summary of the optimization results (8D).

Example index	Objective properties, $[\nu_0, E_0]$	Order of SGA cases	Error, $[\%\nu, \%E]$	Total number of generations
1	[0.1965, 1203.59MPa]	122	$[0.000\%, 0.000\%]$	93
2	[0.1802, 1217.30MPa]	12132	$[0.007\%, 0.001\%]$	174
3	[0.1902, 1181.81MPa]	122132	$[0.004\%, 0.010\%]$	200
4	[0.1684, 1149.97MPa]	1	$[0.001\%, 0.007\%]$	50
5	[0.1838, 1282.16MPa]	121	$[0.006\%, 0.003\%]$	117
6	[0.1783, 1175.81MPa]	12132	$[0.001\%, 0.008\%]$	171
7	[0.1891, 1151.91MPa]	1	$[0.008\%, 0.008\%]$	36
8	[0.1817, 1094.37MPa]	1	$[0.005\%, 0.001\%]$	54
9	[0.1470, 1062.27MPa]	1221	$[0.006\%, 0.007\%]$	140
10	[0.1504, 1144.93MPa]	122	$[0.001\%, 0.010\%]$	101
11	[0.1606, 1101.14MPa]	1	$[0.002\%, 0.010\%]$	47
12	[0.1670, 1235.14MPa]	122131	$[0.007\%, 0.005\%]$	201
13	[0.1826, 1372.53MPa]	1221	$[0.004\%, 0.008\%]$	139
14	[0.1954, 1293.73MPa]	12	$[0.001\%, 0.007\%]$	92
15	[0.2031, 1169.70MPa]	1222	$[0.007\%, 0.003\%]$	141
16	[0.2088, 1212.85MPa]	12	$[0.007\%, 0.006\%]$	74
17	[0.2187, 1119.45MPa]	12132	$[0.007\%, 0.007\%]$	188
18	[0.2031, 1073.87MPa]	12222	$[0.002\%, 0.004\%]$	189
19	[0.1882, 1028.95MPa]	1	$[0.003\%, 0.002\%]$	47
20	[0.1845, 1165.72MPa]	1	$[0.008\%, 0.006\%]$	44

and 8D numerical cases. We define the control group as the same tests processed in Sec. 3.3 and directly refer to Table 3.3 (2D) and Table 3.5 (8D) as the control group's results. In the experimental group, every setting of the GA and the choice of the random point generators are the same as Sec. 3.3 except for the population size and the stopping criteria. The first experimental group maintains a population size of 16. The second experimental group doubles the population size from 16 to 32 in the 2D study case and quadruples the size from 16 to 64 in the 8D study case. Meanwhile, we appropriately adjust the stopping criteria for all the experimental groups, such as the maximum iteration number and the default fitness tolerance. It aims to avoid the optimization process stopping early with unsatisfactory results caused by the relatively strict stopping criteria in Sec. 3.3. Besides, all study cases have the

same error tolerance defined in Sec. 3.3. The errors in the 2D case have an error tolerance, in which both Young's modulus and Poisson's ratio should be less than 0.05%. In the 8D case, the errors should be less than 0.01%. The following tables summarize the comparison results between the control and experimental groups, containing the population size, the stopping criteria, the average error between optimal and objective properties in percentage, the number of counts that the optimal solutions satisfy the error tolerances among the 20 inverse design problems in Sec. 3.3, and the average running time in hours.

In Table 3.6 and 3.7, the first experimental group, with the same population size and the relatively lenient stopping criteria compared to the control group, is unable to achieve the optimal solutions that satisfy the error tolerance defined in Sec. 3.3. Then, we double the population size in 2D case and even quadruple the population size in 8D case. The increases in the population size cause the increased amount of candidate solutions in a single cycle of the SGA, leading to a large possibility of achieving feasible individuals. In Table 3.8 and 3.9, the large population size is found to improve the accuracy in the optimal solutions of SGA. In both 2D and 8D cases, the number of counts in the feasible solutions is increased. Moreover, both cases demonstrate a certain level of improvement in the average errors; however, they still do not satisfy the tolerance. Nevertheless, this improvement causes a substantial growth in the computational expense.

Table 3.6: Comparison of results between the control group and the first experimental group (2D).

Group	Population size	Maximum iteration number	Fitness tolerance	Average error, $[\%\overline{\nu}, \%\overline{E}]$	Counts (satisfied)	Average time (hrs)
Control	16	50	1e-5	$[0.009\%, 0.012\%]$	20	2.49
Experiment #1	16	50	1e-5	$[0.389\%, 0.334\%]$	6	1.74

Table 3.7: Comparison of results between the control group and the first experimental group (8D).

Group	Population size	Maximum iteration number	Fitness tolerance	Average error, $[\%\overline{\nu}, \%\overline{E}]$	Counts (satisfied)	Average time (hrs)
Control	16	50	1e-5	$[0.004\%, 0.006\%]$	20	2.86
Experiment #1	16	50	1e-5	$[0.093\%, 0.084\%]$	3	2.19

Table 3.8: Comparison of results between the control group and the first experimental group in 2D.

Group	Population size	Maximum iteration number	Fitness tolerance	Average error, $[\%\overline{\nu}, \%\overline{E}]$	Counts (satisfied)	Average time (hrs)
Control	16	50	1e-5	$[0.009\%, 0.012\%]$	20	2.49
Experiment #2	32	100	1e-7	$[0.069\%, 0.131\%]$	12	3.42

Table 3.9: Comparison of results between the control group and the first experimental group in 8D.

Group	Population size	Maximum iteration number	Fitness tolerance	Average error, $[\%\overline{\nu}, \%\overline{E}]$	Counts (satisfied)	Average time (hrs)
Control	16	50	1e-5	$[0.004\%, 0.006\%]$	20	2.86
Experiment #2	64	300	1e-7	$[0.004\%, 0.011\%]$	13	16.9

Thus, our proposed inverse design framework has the capability to obtain the optimal solution with high accuracy while maintaining outstanding computational efficiency in solving the inverse design problem with a few objective properties.

3.4.2 Drawbacks of deep learning methods

For a large number of objective properties, a well-trained neural network (NN) definitely has a considerable advantage in computational efficiency over our method. On the other

hand, the users of our approach do not need to build an extensive database of NN that can exhaust computational resources. During the training process of NN, the lack of data limits the design space of the materials while reducing the accuracy of the predicted results. Moreover, we solve the inverse design problem over physically explainable optimization in the geometric patterns of the RVEs. There is no guarantee that the properties of CMMs are smoothly distributed in the material property closure. Therefore, it questions the reliability of regression in deep learning. In an NN, the implicit connections between the input and output layers hinder its versatility and scalability. Instead, our proposed method can deal with different material types by directly modifying the underlying physics-based model in our programming.

Chapter 4

MATERIAL-PROPERTY SPACE EXPLORATION OF POROUS METAMATERIALS

Chapter 4 develops a systematical mapping framework to explore material-property spaces of porous metamaterials using a genetic algorithm (GA) based on the principle of natural evolution. Section 4.1 introduces a sampling method for space exploration and provides a mapping framework to expand an initial property space to its maximal space that contains all possible combinations of properties in a 2D material property chart. The detailed step-by-step procedures for mapping the material property space are demonstrated in Sec. 4.2. The section also checks whether our explored property space is a full closure that reaches its maximum space region. Section 4.3 discusses possible applications of our comprehensive property space explored by the proposed framework. These applications include selecting materials, obtaining alternative solutions for fabrication, and evaluating the applicability of NNs.

4.1 Procedures of Mapping Material-Property Space

This study employs the genetic algorithm (GA) as a sample algorithm to exercise geometry-property models over a microstructure hull consisting of all possible geometric patterns of the RVEs. By using biological evolution, the GA updates geometric parameters relevant to the physical model of the CMMs and eventually produces the geometric patterns, which have the same or similar structural properties as user-defined objective properties. Each child of the GA has a corresponding combination of structural properties. After running the GA with objective properties, the framework collects the children with identical structural properties and scatters them into a material-property chart. The bounded region of these property points (children of the GA) illustrates the explored material-property space during the stochastic searching process of the GA. Unlike the convex hull, this framework defines the region by using a built-in function in MATLAB called *boundary*, which can precisely envelop the points by shrinking towards the interior of the convex hull [66]. Therefore, the first part of the framework uses these property points as prior information to estimate an initial material-property space by piecing up the bounded regions corresponding to various objective properties (Fig. 4.1A). To choose the objective properties, the framework first determines the extreme values of structural properties under consideration, Young's modulus and Poisson's ratio (Fig. 4.1A-1), and then roughly estimates a possible material property space (Fig. 4.1A-2) with a rectangle shape whose side length determined by the extreme values.

After that, the objective properties are determined by uniform sampling (Fig. 4.1A-3) in this rectangular material property space. To precisely estimate the initial material property space of the CMMs, the GA processes the stochastic search algorithm (Fig. 4.1A-4) to check whether the sampled objective properties are within the actual property space of the CMMs.

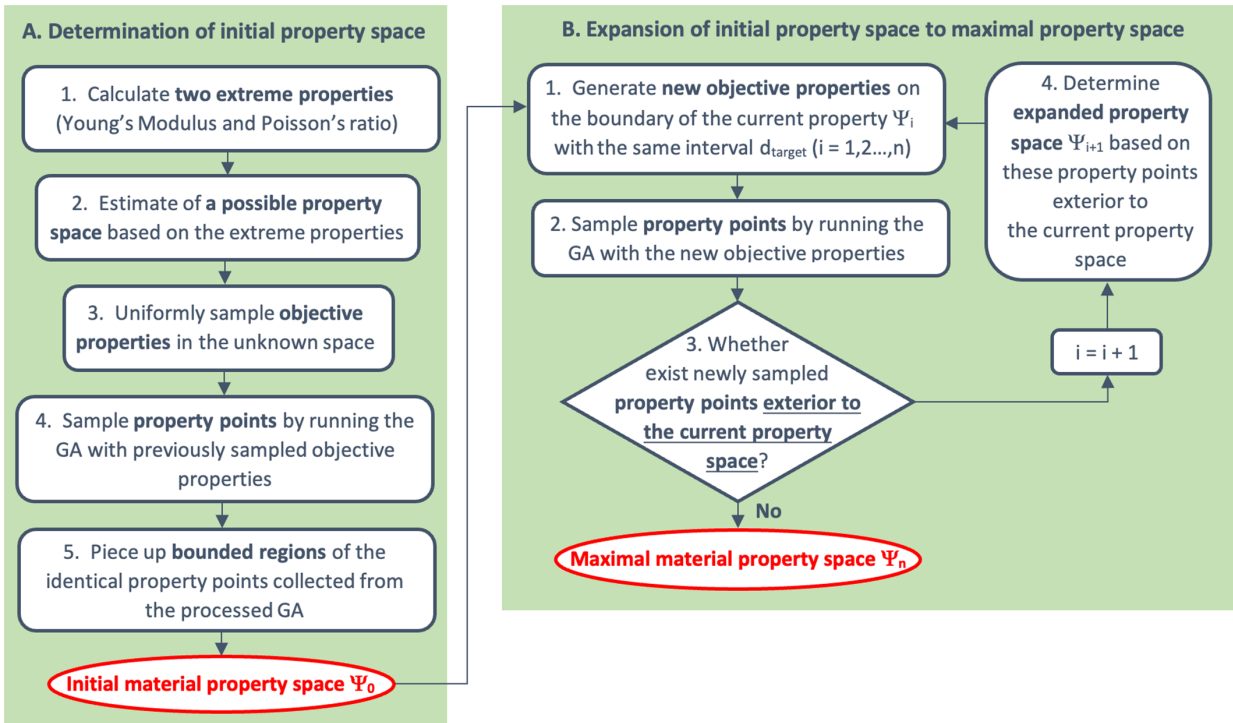


Figure 4.1: Flowchart of sampling algorithm including the selection of objective properties and the stochastic search algorithm of property points.

Consequently, an initial material property space can be determined by the overall space region covered by the convex hulls in a 2D material-property chart (Fig. 4.1A-5). To maximize the region of the initial property space, the second part of the framework selects new objective properties on the explored space boundary to push the newly generated geometric patterns from the GA past the current boundary (Fig. 4.1B). It applies a similar sampling algorithm as the first part but with a different selection method of objective properties (Fig. 4.1B-1). The objective properties are selected on the space boundary, which induces the stochastic search algorithm to generate property points that bypass the current boundary (Fig. 4.1B-2). The algorithm progressively expands the property space (Fig. 4.1B-4) to be maximal until none of the newly generated property points are located outside the current explored space (Fig. 4.1B-3). To validate the maximal region of the explored property space,

new objective properties are generated near and beyond the space. These objective properties attempt to force the GA to search unexplored property spaces that might contain a new combination of properties, which can be physically achieved by undiscovered geometric patterns of the RVEs in the microstructure hull. After a new round of the stochastic searching process, the users can approximate its region to be maximal if none of the children or property points is located exterior to the current maximal space. To illustrate the sampling algorithm of our framework concisely, a flow chart is plotted in Fig. 4.1.

The basic idea of sampling the property points is to solve the inverse design problem of CMMs by the GA. This study formulates the inverse problem into a combined objective optimization problem that aims to minimize the difference between the objective properties and predicted structural properties of the RVEs. By minimizing the difference, the objective properties guide the GA to sample property points near desired locations, which could help to define an initial property space or bypass the explored property space. In this study, the objective properties are the two structural properties of CMMs, Young's modulus and Poisson's ratio. The mathematical formulation of the inverse problem reads as follows:

$$\begin{aligned}
&\text{Minimize}: f(\zeta_1, \zeta_2, ..., \zeta_n) = [w_1(\nu_{12} - \nu_0)]^2 + [w_2(E_{11} - E_0)]^2 \\
&\text{Find}: \zeta = (\zeta_1, \zeta_2, ..., \zeta_n) \\
&\text{Subject to}: \zeta^L \leq \zeta_i \leq \zeta^U (i = 1, 2, ..., n) \\
&\qquad\qquad rsin(\theta) + L_{tolr} \leq \tfrac{L_o}{2},\ rcos(\theta) + L_{tolr} \leq \tfrac{L_o}{2}
\end{aligned} \tag{4.1}$$

where ζ= (ζ_1,ζ_2,...,ζ_n) denotes the vector of design variables and geometric parameters of the CMMs. ζ^L and ζ^U are the lower and upper bounds of the design variables, respectively, and w is a weighted coefficient to rescale the two effective properties in the same numerical level. ν_0 and E_0 are the objective Poisson's ratio and Young's modulus of the GA, respectively. In this study, the poor simulation results from ABAQUS are mainly contributed by failures to

generate feasible geometric patterns of the RVEs. For example, internal pore shapes beyond the RVEs lead to bad connectivity of the structures. Moreover, thin connections between the four symmetrical parts cause a large chance of structural failures under a small strain value. To avoid the above issues, the geometric constraint is assigned as shown in Fig. 3.1. Using this constraint, the smallest distance between the boundary of the pore and the edge of its unit cell should be larger than a length tolerance (L_{tolr}), which is one-twentieth of the side length of the unit cell. If the geometric patterns violate the constraint, the value of the objective function is appended with a penalty value, which is determined based on the level of magnitude of the objective value.

4.2 Results

4.2.1 Preliminary settings

This section presents the numerical results of two families of CMMs, the RVEs with periodical unit cells (PUCs) and with non-periodical unit cells (non-PUCs). The framework includes four main steps: (i) discovering the initial mechanical property chart, (ii) expanding the incomplete material-property space, and (iii) validating the converged material-property space to be maximal in the maximal property space. Here, a case study of RVEs with PUCs demonstrates the detailed procedures of our proposed sampling algorithm for the determination and validation of maximal material-property space. By considering their complex geometric patterns, a case study of RVEs with non-PUCs illustrates the UQ of its maximal property space. In these study cases, the base material of the RVE is chosen as ULTEM® 1000, an amber transparent high-performance polyetherimide (PEI) with isotropic elastic properties, i.e., Young's Modulus of 3580 MPa and Poisson's ratio of 0.36 at room

temperature [67]. For all the study cases, the basic settings of the GA are almost the same to check the compatibility of our sampling algorithm. The same settings of the GA include the stall generation limits, crossover rate, mutation rate, and function tolerance of GA, which are set to 16, 0.8, 0.2, and 10^{-6}, respectively. However, the population size and maximum iteration number vary according to specified factors, such as the number of geometry parameters, locations of selected objective properties, and certain procedures of the sampling algorithm (See details in Sec. 4.2.2). For each objective property, their stochastic search process will be terminated when the relative change in the best fitness function value over the maximum stall generation is less than the tolerance, or the maximum generation number is reached. The major numerical results are illustrated in the following subsections, including convergence histories of material-property spaces of the two families (Figs. 4.6 and 4.10), and maximal/converged material-property spaces of them (Figs. 4.7 and 4.11).

4.2.2 Estimation of the material property space of RVEs with PUCs

Following the sampling algorithm in Fig. 4.1, our proposed framework explores the material-property space of the simplest RVE consisting of periodic unit cells, whose geometric patterns are only controlled by two parameters ζ_i (i = 1,2) shown in Fig. 2.2a. Initially, there is no prior information about the actual property space, consisting of possible property points that can be physically realized by tailoring geometry patterns of the RVEs. Instead of random sampling in RVEs without clues, our framework processes the GA as a sampling algorithm with the objective properties achieved by uniform sampling in a possible property space. The possible property space is a rectangle-shaped region whose side lengths depend on values of extreme structural properties. The framework employs the GA to determine the extreme properties to avoid local convergence in the optimization process due to the

complex parameter domain shown in Fig. 4.2a [44]. In this case, the objective function in Eq. 4.1 is replaced by E_{11} or ν_{12} for the minimization (minimize $-E_{11}$ or $-\nu_{12}$) and for the maximization with the same constraint. Given the population size defined as 32 and the

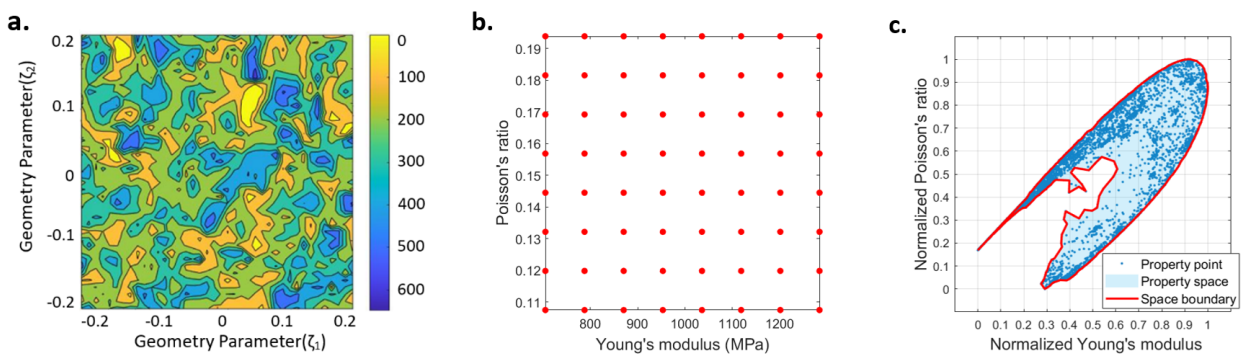


Figure 4.2: Determination of initial material-property space: (a) contour of objective values in 2D parameter space, (b) uniform sampling of objective properties in a possible property space, and (c) initial material-property space (normalized).

maximum iteration number defined as 150, the ranges of Young's modulus and Poisson's ratio are estimated by the GA, which are [704.62, 1284.58] MPa and [0.1074, 0.1939], respectively. To standardize the data and make it easy to analyze, structural properties generated by the current and following steps of the expansion process are all normalized using their extreme values determined in the current step. In Fig 4.2b, 64 property points are uniformly sampled in this rectangular-shaped property space and are defined as objective properties of the GA. The stochastic search process samples property points surrounding these objective properties and checks whether the objective properties are located in the actual property space. Thus, this framework applies GA not only to estimate the region of the material-property space in longitudinal and transverse directions based on the extreme properties but also to estimate the region from all other possible directions. For the sampling purpose, the population size is reduced to 16, with the maximum iteration number defined as 100 in Step A-4 (Fig. 4.1). It also helps to reduce the computational cost. The average errors obtained by the

experiments between the objective and predicted properties are 0.45% and 0.47% for Young's modulus and Poisson's ratio, respectively. These errors are acceptable because the primary purpose is not achieving the global optimum. The searching algorithm gives 6406 property points (denoted as blue dots in Fig. 4.2c) containing unique combinations of two structural properties. Ultimately, the initial material-property space is determined by piecing up the bounded region of these property points or the explored spaces of each run of the GA. Fig. 4.2c shows the initial material-property space (denoted as a light blue region) illustrated by a normalized material-property chart, whose longitudinal and transverse axes are assigned with Young's modulus and Poisson's ratio, respectively. The region of the space is defined by its boundary, denoted as a red solid line in Fig. 4.2c.

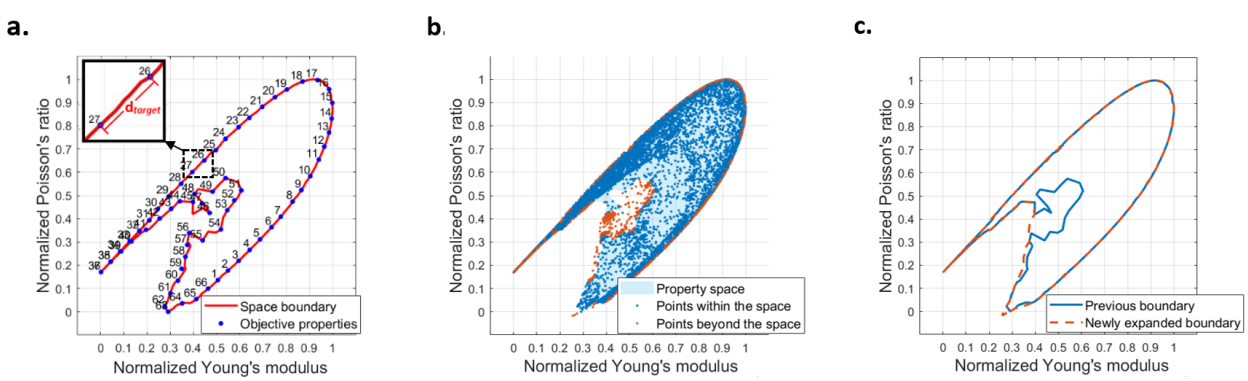


Figure 4.3: The first expansion loop: (a) objective properties selected on the boundary of the initial property space, (b) distribution of all newly generated property points, and (c) comparison between the previous and the newly expanded boundary of the property space.

However, the initial property space shown in Fig. 4.2c is incomplete and fails to cover all possible combinations of structural properties. To this end, this study introduces an intuitive approach to expand the space region to the maximum progressively. This approach begins with selecting 66 new objective properties on the boundary of the initial property space, denoted as blue dots in Fig. 4.3a. Two neighboring objective properties have the same interval, or the shortest Euclidean distance between them, $d_{target} = 0.06$ shown in Fig. 4.3a.

Since the expansion process intends to discover property points (children) located outside the current space region, this framework selects objective properties on the boundary instead of those selected within the space due to the distribution characteristics of the GA's children. After processing the 66 runs of the GA, the sampling algorithm generates 8801 new property points in total. In Fig. 4.3b, all the points are scattered on the initial material-property chart. They are denoted as the blue and orange points, representing the properties within and beyond the initial space, respectively. To expand the initial space, the framework is only interested in the non-overlapping regions between the bounded region of the newly generated points and the initial property space. The non-overlapping region is the newly expanded region of the property space. By combining the region with the initial property space, a newly expanded material-property space is achieved with enriched property points. Fig. 4.3c compares the regions between the newly expanded and initial property spaces, bounded by a dashed orange line and a solid blue line, respectively.

Table 4.1: Results of the GA with the sample violated objective properties.

Objective properties #	# Generations in total	Generation # (violated)
1	56	2, 5, 13, 14, 22
4	45	2
13	52	1, 4
19	50	21, 22
24	48	6, 14, 16, 18, 22
28	62	13
34	52	2, 11, 12, 13, 14, 17
37	47	2, 6, 10
52	46	1
58	46	12, 13

Following the iterative loop in Fig. 4.1B, this framework continues to expand the material-property space by using the sampling algorithm. In the study case, each iterative loop requires processing a large number of stochastic search processes corresponding to numerous objective properties selected on the boundary. With computationally expensive simulations

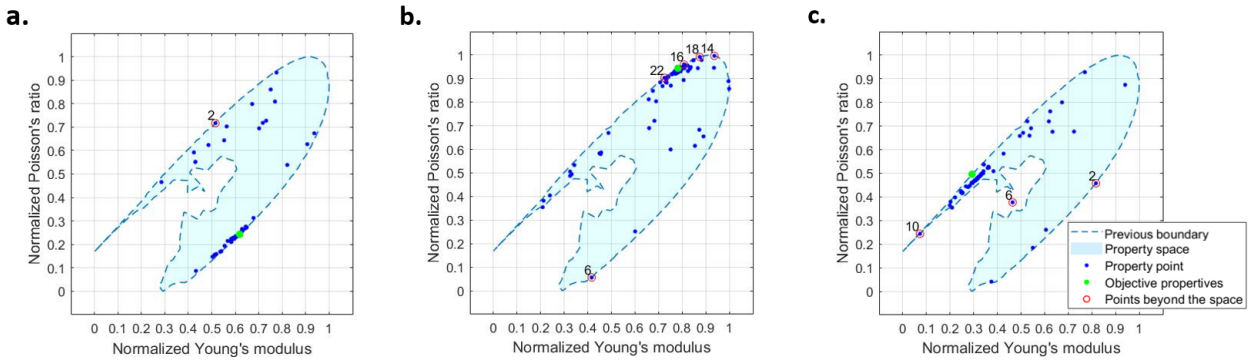


Figure 4.4: Distribution of possible solutions of the sample violated objective properties with the indexes (a) #4, (b) #24, and (c) #37 from Table 4.1.

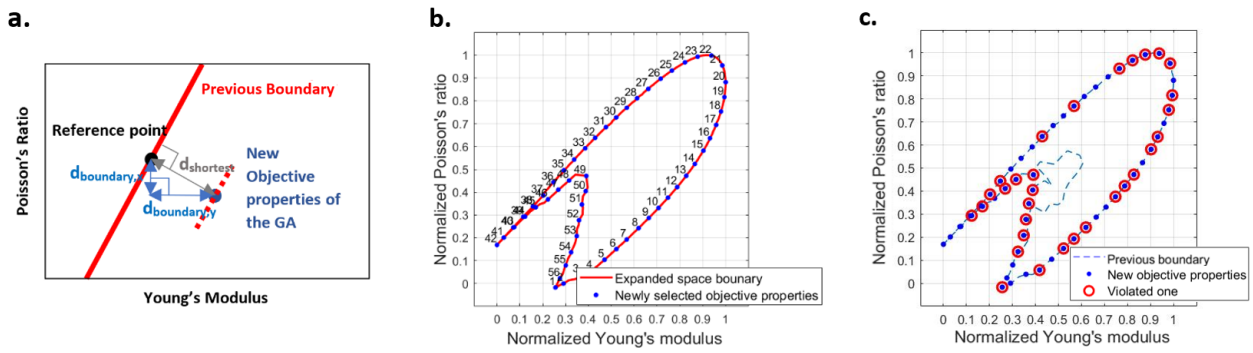


Figure 4.5: Distance constraint applied in the second expansion loop: (a) Evaluation of distance from new objective properties to the previous boundary, (b) objective properties selected on the newly expanded boundary determined in the first expansion loop, and (c) violated objective properties circled in red.

in ABAQUS, the execution time of the single loop is enormous. To increase the computation efficiency of this iterative search process, this study discovers the factors that increase the possibility of discovering property points exterior to the space. After testing with different objective points selected on the newly expanded boundary, the study identified that the point distribution is mainly affected by the shortest Euclidean distance ($d_{boundary}$) between objective properties selected on the expanded boundary (bounded by a dashed orange line in Fig. 4.3c) and the boundary determined from the previous loop/step (bounded by a solid blue line in Fig. 4.3c). Table 4.1 summarizes the experimental results of sample objective

properties, which are close to the previous boundary. It summarizes the GA results, including the total number of generations and the indexes of the generations where the property points are located outside the newly expanded space. The experimental results can be visualized in Fig. 4.4, which illustrates the distribution of property points for three sample objective properties from Table 4.1. To assess how far the objective properties are from the previous boundary, this framework calculates the shortest Euclidean distance, $d_{shortest}$, from the objective properties to the previous boundary. The distance is determined in both longitudinal and transverse directions. In a normalized space shown in Fig. 4.5a, the horizontal and vertical distances of $d_{shortest}$ are denoted as $d_{boundary,x}$ and $d_{boundary,y}$, respectively. Consequently, the distance constraint is defined as:

$$d_{boundary,x} \geq 0.05 \tag{4.2}$$

$$d_{boundary,y} \geq 0.05 \tag{4.3}$$

The distance constraint is violated, or the objective properties are close to the previous boundary if $d_{boundary,x}$ and $d_{boundary,y}$ of the objective properties are both less than 5% of the normalized ranges of two properties. The range is equal to one since it is defined by extreme properties from step A-1 (Fig. 4.1). The distance constraint begins to apply in the second expansion loop. In this loop, 56 new objective properties are selected on the newly expanded space boundary (Fig. 4.5b). Among these, 29 objective properties violate the distance constraints. They are processed in the GA with the reduced maximum iteration number. The violated objective properties are circled in red, shown in Fig. 4.5c. From Table 4.1, the maximum iteration number can be significantly reduced from 150 to 25. In the section on numerical results, this framework employs the same distance constraint and reduced maximum iteration number for the following expansion loops of both RVEs with

PUCs and non-PUCs.

Table 4.2: Summary of the whole space exploration process of the CMMs with PUCs.

Step #	# Objective properties (total)	# Objective properties (violated)	# Property points
0	64	-	6404
1	66	0	8801
2	56	29	5609
3	57	45	4400
4	57	57	3899

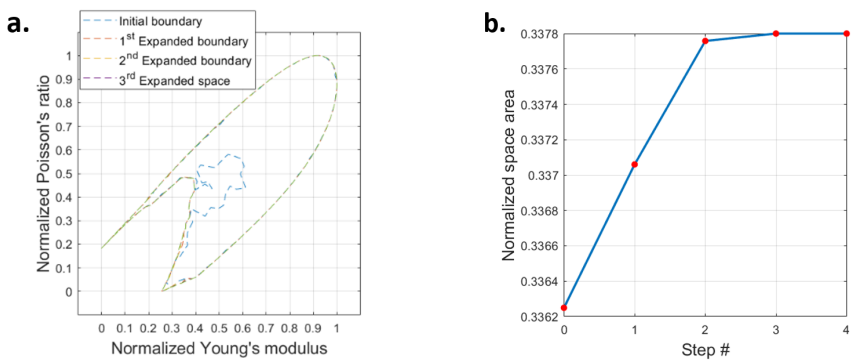


Figure 4.6: Convergence histories of (a) the normalized property space boundary and (b) space area.

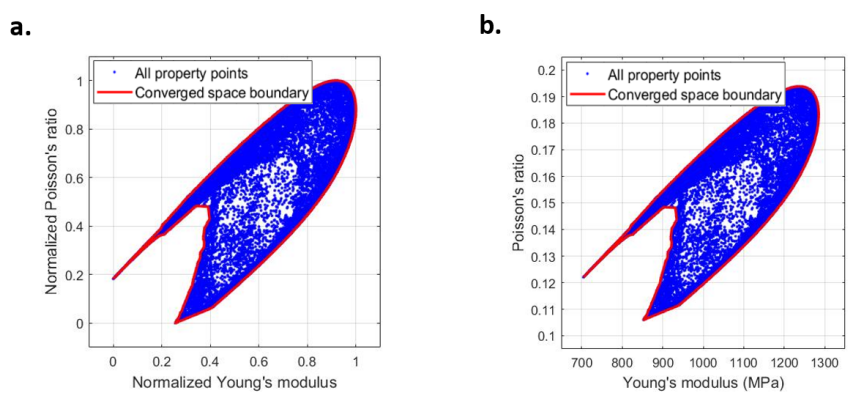


Figure 4.7: Converged property space: (a) the normalized space and (b) the original space.

After three expansion loops, the material-property space converges. In the fourth expansion, none of the newly generated property points is exterior to the previous space boundary. In

Table 4.2, the number of violated objective properties is the same as the number of objective properties selected on the space boundary determined in the fourth expansion. It indicates that the space barely expands or does not expand in the last expansion loop, which also implies the convergence of the space. The numerical results of the whole space exploration process are summarized in Table 4.2. It contains the number of total objective properties, violated objective properties, and newly generated property points in each step. Convergence histories of the normalized space boundary and space area are shown in Fig. 4.6a and Fig. 4.6b, respectively. In summary, 29113 property points are produced through the whole space exploration process, denoted as blue dots in the converged property space shown in Fig. 4.7.

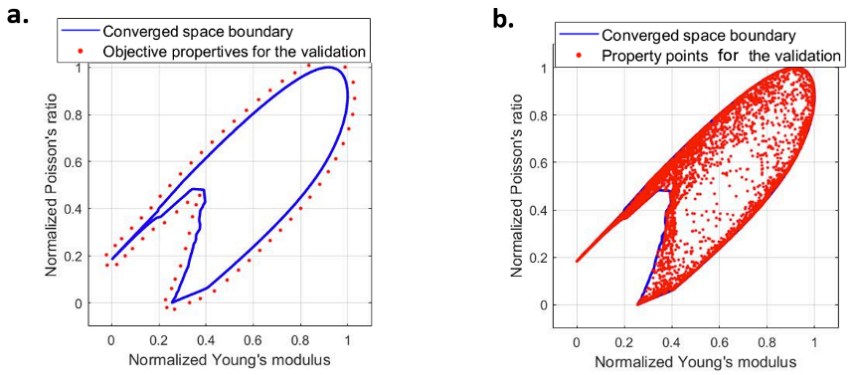


Figure 4.8: Post process of the space exploration: (a) objective properties for the validation and (b) newly generated property points for the validation.

In the post-process, to validate that the converged property space is maximal, 62 new objective properties with the same shortest Euclidean distance from its boundary, $d_{validation}$ = 0.03, are generated in an outward direction that is normal to the final normalized space boundary as shown in Fig. 4.8a. We assumed that the GA has the capability of creating several possible property points that approach these new objective properties and prove that these objective properties are achievable and within the actual property space. By considering that the number and diversity of the children might restrict the expansion of the material-property space, the population size and the maximum iteration number are

increased to 32 and 150, respectively. However, after running the GA, all new 6669 property points cluster near the boundary of the converged property space in Fig. 4.8b and cannot bypass the boundary. It indicates that the assumption is invalid. Thus, the converged material-property space of RVE with PUCs is validated, which has the maximal space region that covers all possible combinations of Young's modulus and Poisson's ratio as shown in Fig. 4.7.

4.2.3 Estimation of the space of RVEs with non-PUCs

The framework is also applied to explore the material property space of CMMs with non-PUCs, whose geometric patterns are controlled by a geometric matrix containing eight geometric parameters $[\zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5, \ \zeta_6, \zeta_7, \zeta_8]$ shown in Fig. 2.2b. Following the same exploration procedure shown in Fig. 4.1, the extreme properties are first determined by the GA. Compared with the parameter space in the previous study case, its dimension in this study case increases from 2D to 8D due to increases in the complexity of geometry patterns of the RVEs. In a high-dimensional parameter space, premature convergence is more likely to appear during the stochastic searching process. To maintain the performance of the GA, the population size and the maximum iteration number are increased to 64 and 250, respectively. With these settings of the GA, the longitudinal and transverse bounds of the property space are determined as $E \in [731.42,\ 1244.40]$ MPa and $\nu \in [0.0904,\ 0.2153]$. Once the rectangular-shaped possible property space is defined, this framework uniformly samples 64 objective properties in the space, illustrated in Fig. 4.9a. For exploring an initial property space, these objective properties are applied to sample property points of the CMM with non-PUCs, which contain combinations of properties that can physically be realized by their corresponding geometric patterns of the RVEs. For the sampling algorithm, the new population size and maximum iteration number are defined as 32 and 150, respectively.

It allows the framework to discover 21481 property points. The average errors for Young's modulus and Poisson's ratio are 0.41% and 0.48%, respectively, which are assumed to be in acceptable ranges for sampling purposes. In Fig. 4.9b, the initial property space is achieved by combining the bounded regions of these sampled points.

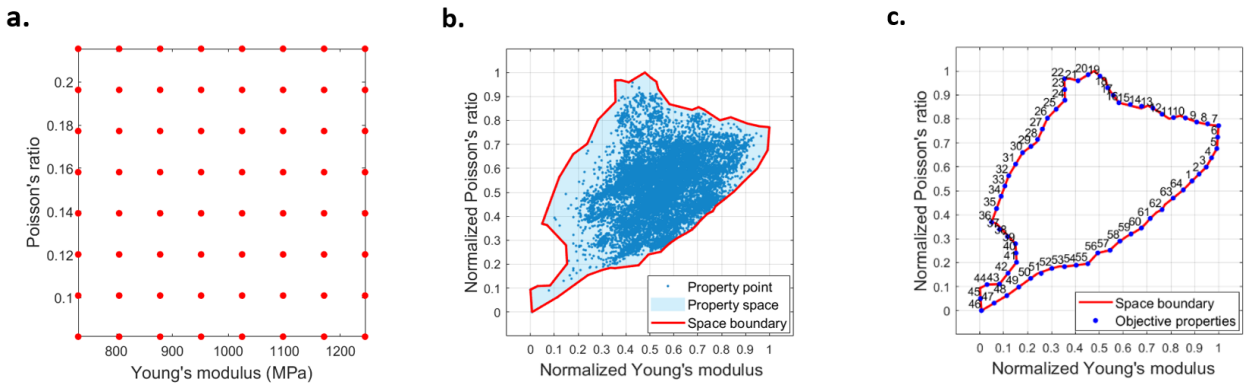


Figure 4.9: Determination of the material-property space: (a) uniform sampling of objective properties in a possible property space, (b) initial material-property space (normalized), and (c) objective properties selected on the space boundary.

To check the compatibility of the framework, the settings of the following iterative expansion loops are the same as in the previous study case. The settings include $d_{target} = 0.06$ for selecting the objective properties on the boundary (Fig. 4.9c) and tolerance = 0.05 for defining the distance constraint. The property space eventually converges after expanding the space region eight times. Convergence histories of the normalized space boundary and space area are shown in Figs. 4.10a and 4.10b, respectively. Combined with property points in estimating initial property space, 94206 unique combinations of Young's modulus and Poisson's ratio are densely scattered in the converged space during the whole space exploration process of the CMMs with non-PUCs (Figs. 4.11a and 4.11b). Table 4.3 summarizes the numerical results of the whole process. In Fig. 4.10a, it is noticed that the longitudinal and transverse bounds of the property space significantly enlarge, as well as the bounds in the other directions. From Fig. 4.11b, the extreme values of Young's modulus and Poisson's

ratio increase to [717.37, 1272.83] MPa and [0.0867, 0.2547], respectively.

Besides, the converged property space is required to be validated as the maximal space. With $d_{validation} = 0.03$, 52 new objectives properties are picked near its converged boundary for processing the validation step (Fig. 4.12a). Considering the expansion limits caused by the limited diversity and quantity of the property points, the population size and the maximum iteration number are increased to 64 and 250, respectively. After all, in Fig. 4.12b, the converged property space is approximately overlapped with the distribution of the new 34893 property points generated in the validation step. However, none of the newly generated property points can bypass the converged space boundary. Thus, the converged property space (Fig. 4.11) is successfully validated as the maximal space.

Table 4.3: Summary of the whole space exploration process of the CMMs with non-PUCs.

Step #	# Objective properties (total)	# Objective properties (violated)	# Property points
0	64	-	21481
1	64	0	14703
2	57	8	12605
3	52	26	8851
4	50	29	8240
5	52	33	8005
6	50	33	7883
7	50	38	6442
8	50	47	3564
9	50	50	2432

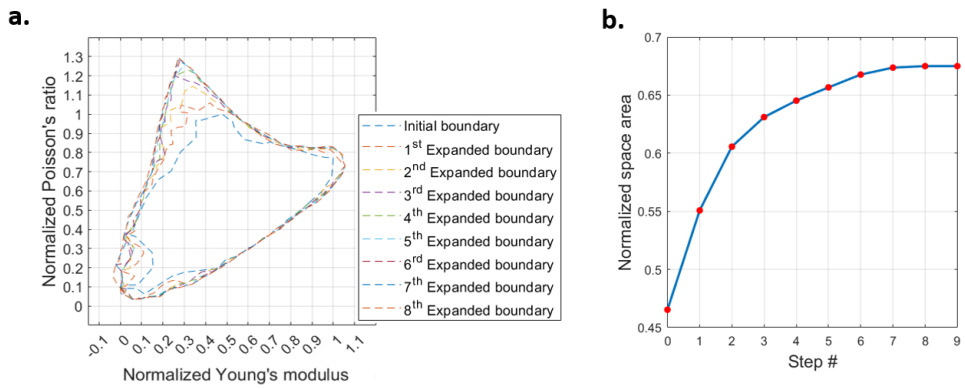


Figure 4.10: Convergence histories of (a) the normalized property space boundary and (b) space area.

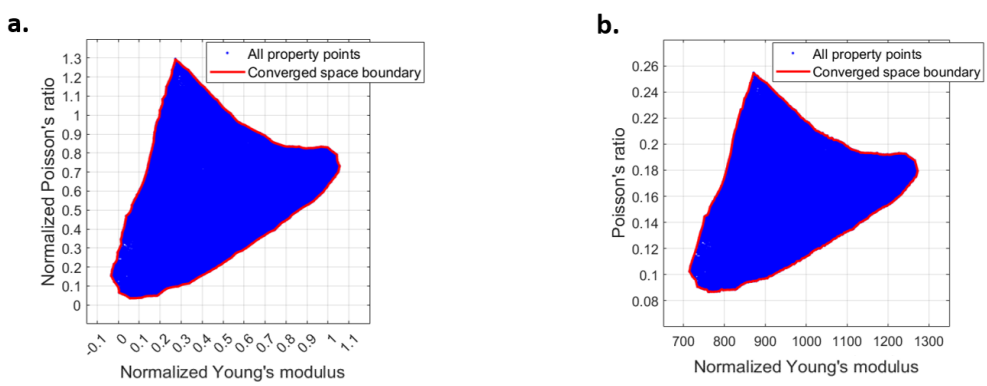


Figure 4.11: Converged property space: (a) the normalized space and (b) the original space.

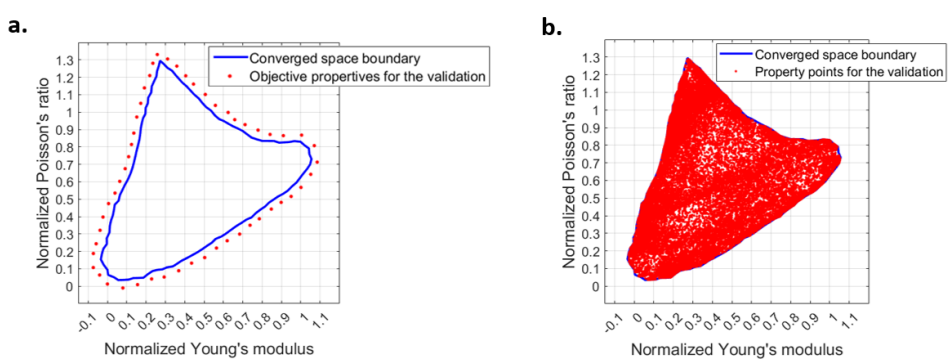


Figure 4.12: Post process of the space exploration: (a) objective properties for the validation and (b) newly generated property points for the validation.

4.3 Discussion

The numerical results from the previous section demonstrate that our proposed methodology can precisely determine the maximal material-property space of the CMMs. However, the major drawback of our methodology is the high computational expense of the whole exploration process due to the stochastic searching algorithm. For example, exploring and validating the property space of RVEs with PUCs requires the total number of selected objective properties to be as high as 362. These objective properties contribute to 362 runs of GA to process the sampling algorithm and complete the exploration process. Although time-consuming, our methodology is not content with a rough approximation of the property space. Instead, with the iterative searching algorithm, our methodology continuously expands the property space and approaches its maximum. Moreover, the mapping procedure is only required once for a specific type of metamaterial.

4.3.1 Material selection

In engineering design, material selection charts are a simple and quick way of initially screening materials for a particular application [68]. In this study, a material selection chart can be achieved by plotting all resulting material-property spaces obtained through our methodology in the same 2D material-property chart shown in Fig. 4.13. Our material selection chart has similar functions as Ashby's charts [69]. To assess materials' suitability for a given application, the chart screens out materials based on the coverage corresponding to their space regions in the same selection chart. In our study scenario, the space region of RVEs with non-PUCs (bounded with a blue line) is much more comprehensive than that of RVEs with PUCs (bounded with a green line) as illustrated in Fig. 4.13. For RVEs with non-PUCs, the wide coverage of their space region is due to their complex geometric patterns

containing various combinations of unit cells. In material applications, a narrow space region (RVEs with PUCs) requires trade-offs between two material properties to achieve users' requirements [68]. An expansive space (RVEs with non-PUCs) offers more options in combination with two properties. It allows its family of materials to have a broad application in engineering design. Moreover, in Fig. 4.13, the overlapping area of the charts (bounded with a red dotted line) provides alternative solutions in manufacturing processes. In this area, RVEs with PUCs share similar mechanical performance as RVEs with non-PUCs but possess simple geometric patterns of RVEs, which are easy to fabricate from a manufacturing point of view. Besides, in Ashby's chart, experimental data for each material family are clustered into multiple regions roughly shaped as bubbles [69]. With our methodology, the space region not only approximately reaches the maximal but is also precisely enveloped with a well-defined boundary. The boundary is initially determined based on a convex hull containing all the feasible property points from the exploration process. It can further shrink towards the hull's interior to represent the space region more precisely. It ensures the accuracy and reliability of material selection. To further proceed with the material selection process, the screening of materials can be followed by ranking the candidate materials using the analytical hierarchy process as the main Multi-Criteria Decision Making methodology for specific manufacturing processes [70].

4.3.2 Applicability of neural networks

The deep learning method opens new doors to designing micro-structured materials effectively by replacing the computationally expensive finite element simulations. Since the neural network (NN) only extrapolates well within the data domain defined by its database for the training process [71], the coverage of the data domain is important for the applicability of the NN. In materials design, the database usually contains pairs of geometry and properties. Its

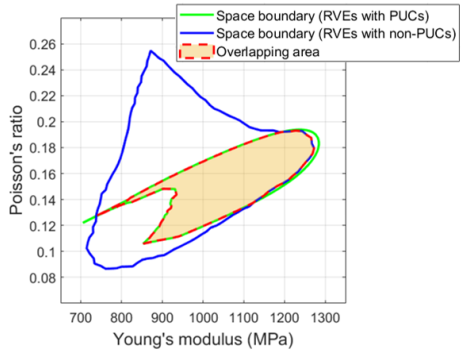


Figure 4.13: Material selection chart of two families of the CMMs including RVEs with PUCs and RVEs with non-PUCs.

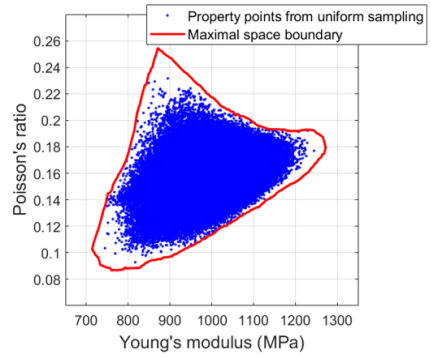


Figure 4.14: Distribution of property points from the uniform sampling and the maximal material-property space boundary from Sec. 4.2.3

data domain of properties can be represented as a material-property space with properties plotted along each axis [72]. Therefore, the applicability of the NN can be represented by a material-property space containing all property points from the database. In a forward fashion, the maximal space boundary (denoted as a red line in Fig. 4.13) is treated as a reference to assess the applicability of the NN in predicting material properties. Theoretically, the wider the data domain covers in the maximal material-property space, the better the applicability of its NN is for a specific class of metamaterials defined in the database. Besides, although there has been research about advances in various deep-learning methods for material design problems, database generation has not been thoroughly investigated. A common method of database generation is the uniform sampling of design variables. To compare with our sampling algorithm, a new database of RVE with non-PUCs is built using uniform sampling with the same number of property points, which retrieves a maximal property space. Due to the central limit theorem, most of them (denoted as red dots in Fig. 4.13) become concentrated around a central position of their distribution and restrict the spread of the data to reach its peripheral unexplored material-property space. In the periphery of the data domain, the lack of data might induce bias and errors during the

training process of the NN. On the contrary, during the iterative expansion process, our sampling algorithm continuously generates property points (denoted as blue dots in Fig. 4.10) to fulfill the region near the space boundary. With the K-function and down-sampling method, our database becomes more evenly distributed and alleviates the issue mentioned above. Moreover, with the same number of design samples, our sampling strategy provides wide coverage of the data domain, which provides better applicability for the NN compared to the domain achieved by uniform sampling.

Chapter 5

MATERIAL-PROPERTY SPACE EXPLORATION OF POROUS METAMATERIALS UNDER UNCERTAINTY

Chapter 5 investigates variations in material property spaces of porous metamaterials induced by manufacturing defects. This investigation is the extension of the study in Chapter 4. This chapter mainly focuses on the manufacturing defect called surface distortion. The defect is mimicked as the uncertainty of the geometric variables as illustrated in Sec. 5.1. In Sec. 5.2, the study quantifies the stochastic properties, associated with the surface distortion, using Monte Carlo sampling with an NN-based surrogate model. The uncertainty of the properties is visualized through confidence intervals and variations of the space boundary in Sec. 5.3. Section 5.4 discusses the effects of surface distortion on the effective Young's modulus and Poisson's ratio values of porous metamaterials.

5.1 Realization of Surface Distortion

This study aims to quantify the effects of the uncertainty on the material-property space of the RVEs with non-PUCs. Here, the uncertainty of properties is assumed to result from the surface distortion occurring during a fused filament fabrication (FFF) process [73]. For FFF-printed polymer parts, the defect occurs due to uneven thermal shrinkage, which is induced by poor heat conduction and residual stresses [74, 75]. This study mimics the manufacturing defect by varying the porous shapes within RVEs. To simplify the uncertainty analysis, this study characterizes the variations using the eight design parameters of the geometric shape matrix shown in Fig. 2.2b. As illustrated in Fig. 5.1, a minor change in the parameter values allows for slight expansion and shrinkage of the porous shapes from different polar angles (θ in Eq. 2.1). By referring to the tolerances of FFF printing [74, 76], the uncertainty levels of the geometric parameters are specified in three cases with the following tolerance values: $d\zeta_i \sim U(-10\%, +10\%)$, $d\zeta_i \sim U(-20\%, +20\%)$, and $d\zeta_i \sim U(-30\%, +30\%)$. With the fixed side length of the RVEs, $L = 10$ mm, the corresponding tolerances for the polar radius (r) are assigned: $\pm 0.09mm$, $\pm 0.19mm$, and $\pm 0.28mm$.

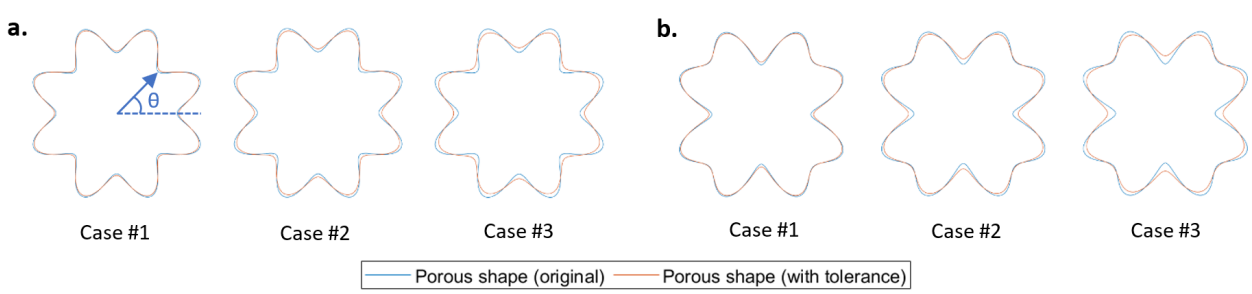


Figure 5.1: Variations in the porous shapes are defined through three tolerance cases ($\pm 10\%, \pm 20\%, \pm 30\%$) with corresponding shape parameters: (a) $[\zeta_1, \zeta_2] = [0.01089, -0.2012]$ (b) $[\zeta_1, \zeta_2] = [-0.1598, -0.1797]$.

5.2 Uncertainty Quantification (UQ) of Structural Properties

Owing to its simplicity and robustness when working with high-dimensional design spaces (Fig. 2.2b, the Monte Carlo (MC) method is utilized to quantify various outcomes for structural properties under the effects of the uncertainty arising from manufacturing defects [77]. In Fig. 5.2a and 5.2b, after 50,000 MC iterations, satisfactory convergence behavior is obtained for the first (mean, m_1), second (standard deviation, m_2), third (kurtosis, m_3), and fourth (skewness, m_4) statistical moments for both Young's modulus and Poisson's ratio. The distribution histograms of the two properties shown in Fig. 5.2c suggest that the two properties can be approximately considered as normally distributed random variables. In Fig. 5.2c, the geometry constraint causes a lack of samples of Young's modulus in the left tail of its histogram. Based on the satisfactory convergence behavior, our study processes MC sampling with 50,000 samples for 120 representative property points selected on the maximal space boundary (Fig. 5.4). These property points represent the contour of the maximal space region and contain information on the deterministic geometry matrix consisting of eight parameters. In the aforementioned three tolerance cases, the geometric matrix contains stochastic geometric parameters, which demonstrate uniform distributions and contribute to the variations in their geometric patterns. Consequently, the variations lead to uncertainties in the structural properties. The combination of these uncertainties alters the deterministic region of the maximal material-property space.

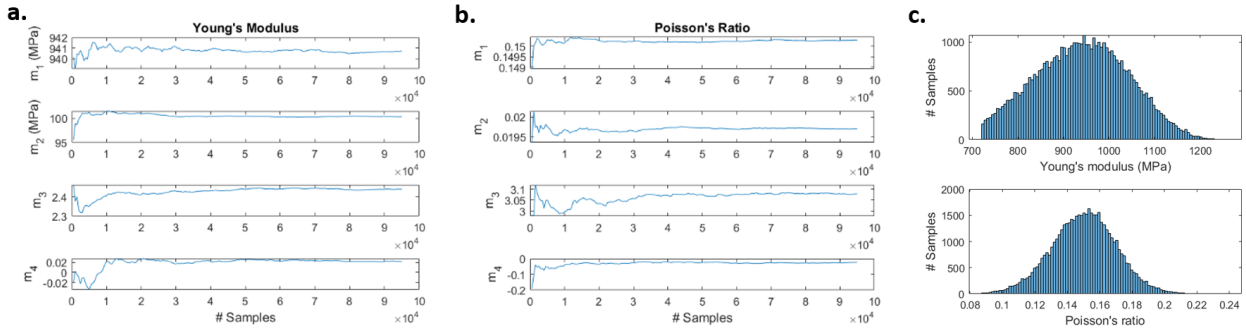


Figure 5.2: Pre-processing of the MC method: (a) convergence histories of statistical moments for Young's modulus, (b) convergence histories of statistical moments for Poisson's ratio, and (c) distribution histograms of the two properties.

5.3 Results

Here, the deep learning method is applied to accelerate the MC sampling. For numerous samples of the geometric parameters, it helps to predict the corresponding structural properties in a forward fashion efficiently. The database is built by collecting all the property points throughout the space expansion and validation process in Sec. 4.2.3. It consists of 129099 pairs of geometric matrices and their related properties, including Young's modulus (E) and Poisson's ratio (ν). In the database, the property distribution is shown in Fig. 5.5a. However, the convergent behavior of the GA's children induces a specific pattern appearing in the distribution of samples from the database, such as the clusters with high sample density shown in Fig. 5.3b. It can introduce error or bias to the outcomes of an NN trained with this database. To prevent this potential issue, sample data in these clusters are partially eliminated to achieve dispersed distribution in the data domain (Fig. 5.3c). It is performed by evaluating sample density with Ripley's K-Function, a tool for analyzing point patterns in a 2D space [78]. After downsampling the samples in these clusters, the modified database remains with 73204 geometry-property pairs showing an approximate uniform distribution in Fig. 5.5b. This study splits the database into 80% of it for training and 20% for testing.

Based on their extreme values, all data in the database are normalized to a common scale from 0 to 1. Our NN is devised into Multi-layer perception (MLP) architecture, constructed as two fully connected layers with 25 neurons (Fig. 5.6a). It is developed and tested using Keras with a TensorFlow backend and optimized in terms of the mean square error (MSE) metric. A regularization technique called early stopping is applied to prevent overfitting, which allows the training process to run for 37 epochs in total. The training process takes approximately 5.5 minutes on a single tensor processing unit version 3.0, which has 16GB of high-bandwidth memory pre-core. After the 37 epochs, the MSE is minimized to 1.7752e-04 and 1.7815e-04 for the training and validation datasets, respectively. Their small values and stable convergences (Fig. 5.6b) indicate the excellent capability of the NN to predict Young's modulus and Poisson's ratio as a function of user-defined design parameters. Furthermore, with 26684 samples from the test dataset, the capability can also be proven by showing the actual properties from the database and the predicted properties from the NN in the same plot. In Fig. 5.6c, a good fitness in the bisection line and narrow bandwidth of the scatter distribution validate the reliability of our NN. The good fitness can be numerically represented as the coefficient of determination (R^2) values of 0.9963 for E_{11} and 0.9973 for ν_{12}.

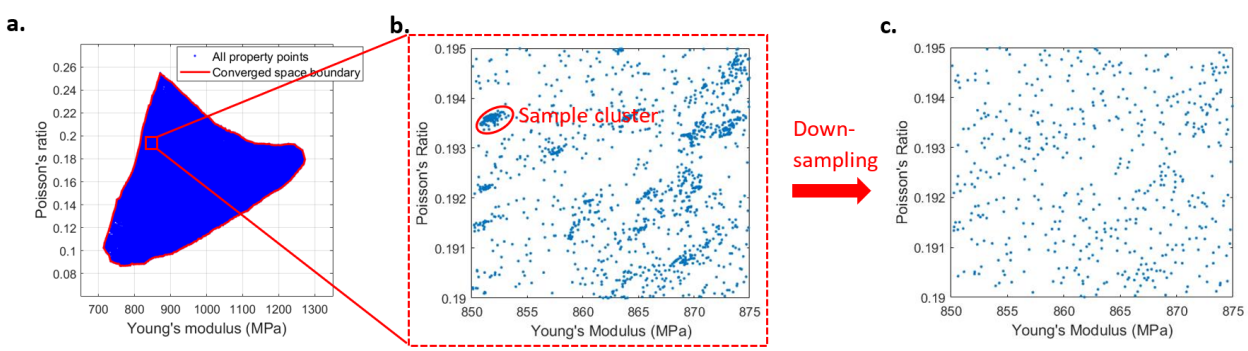


Figure 5.3: Modification of the database: (a) the maximal material-property space, (b) detection of the clusters with high sample density, and (c) downsampling of samples in the detected clusters.

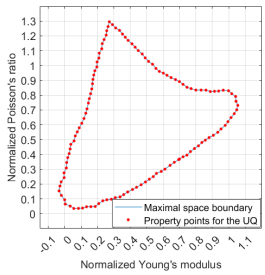


Figure 5.4: Property points for the UQ.

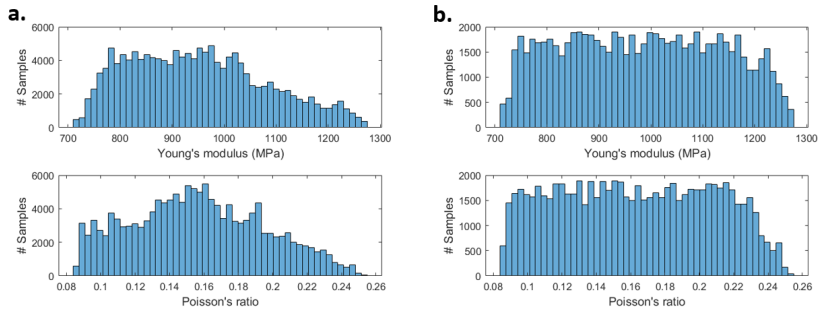


Figure 5.5: Distribution histograms of (a) original and (b) modified database.

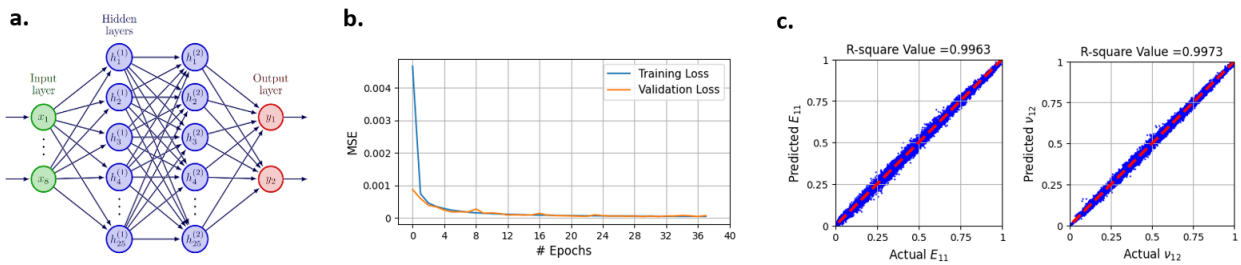


Figure 5.6: Deep learning method: (a) architecture of the NN, (b) convergence history of the MSE, and (c) comparisons between the actual and predicted properties.

Once the NN is well-trained, it can accelerate the MC sampling and efficiently predict structural properties associated with stochastic geometric parameters from three tolerance cases. To quantify the stochastic responses in the statistical sense, the mean values of two properties over the pre-defined uncertainty range and corresponding variances are obtained as follows:

$$\overline{x}_{prop}(\zeta_i) = \frac{1}{N}\Sigma_{i=1}^{N} x_{prop,i}(\zeta_i) \tag{5.1}$$

$$\sigma_{x_{prop}}(\zeta_i) = \sqrt{\frac{\Sigma_{i=1}^{N}(x_{prop,i}(\zeta_i) - \overline{x}_{prop}(\zeta_i))^2}{N}} \tag{5.2}$$

where N is the size of MC samples, ζ_i is the geometric matrix of stochastic design parameters, and x_{prop} denotes the structural properties under uncertainty. The 95% confidence interval

of the structural properties under uncertainty can be determined as:

$$x_{prop,95\%}(\zeta_i) = \overline{x}_{prop}(\zeta_i) \pm 2\sigma_{x_{prop}}(\zeta_i) \tag{5.3}$$

For the 120 selected property points, their mean values and corresponding variances are shown as 95% confidence intervals with error bars in Figs. 5.7a and 5.7b. By combining the interval regions of the two properties, the uncertainty in material-property space is determined. It can be visualized as variation intervals of the space boundary in the 2D material-property chart shown in Fig. 5.7c. Also, the uncertainty in combinations of two properties can be numerically represented as variations in the maximal space area shown in Table 5.1 with the three tolerance cases.

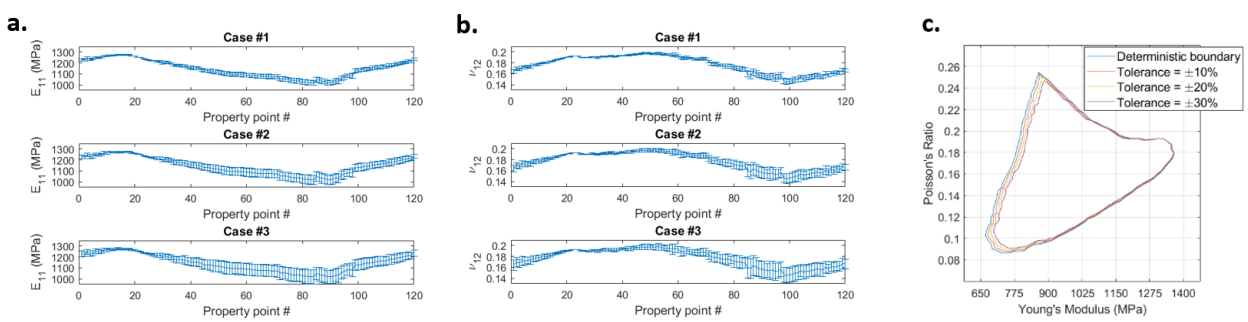


Figure 5.7: Variations in properties associated with the three tolerance cases: (a) error bars for Young's modulus of all the property points for the UQ, (b) error bars for Poisson's ratio, and (c) variation intervals of the maximal space boundary.

Table 5.1: Summary of normalized areas under the three tolerance cases.

	Case #0 (±0%)	Case #1 (±10%)	Case #2 (±20%)	Case #3 (±30%)
Normalized area	0.6750	0.6484	0.6228	0.5963

5.4 Discussion

With advances in additive manufacturing resulting in the proliferation of metamaterials, an evaluation of their tolerances and defects arising from manufacturing is vital. Deviations from the expected geometry are possible and they may alter the mechanical properties of printed parts. This study mainly investigates the uncertainty in the form of surface distortion arising during the FFF printing process and its impacts on the structural properties of the CMMs. In Fig. 5.7c or 5.8a, the variation intervals of the maximal space boundary allow for visualizing the impacts on combinations of the two properties under consideration, Young's modulus (E_{11}) and Poisson's ratio (ν_{12}). Both Fig. 5.7c or 5.8a and Table 5.1 indicate an approximately linear relationship between variations of the maximal space area and tolerances of the geometric variables. In general, as the level of uncertainty of design parameters increases, the confidence interval of properties is expanded (Figs. 5.7a and b), leading to a larger shrinkage of the property space (Fig. 5.7c or 5.8a). Besides, the variation interval expands as Young's modulus decreases as shown in Fig. 5.7c or 5.8a. It indicates that the defect has less impact on the two properties of RVEs with high E_{11}. The complexity of geometric patterns shown in Fig. 5.8a demonstrates the extent of the impact causing the variations in properties. As Young's modulus decreases, the complexities of the pore shapes in RVEs increase, leading to higher uncertainty of mechanical properties. Here, this study further investigates an individual property instead of the combination of two properties. This study captures the effect of manufacturing defects on an individual property not only based on the property points on the boundary (Fig. 5.8a) but also located within the property space (Fig. 5.8b). In Fig. 5.8b, seven representative property points are selected in the space. Their variation intervals and corresponding geometric patterns indicate a similar trend of uncertainty along with the axis of Young's modulus. Besides, for the selected points in Figs. 5.8a and b, Table 5.2 summarizes the average percentage differences (%diff) between

their deterministic properties and properties under the effects of the uncertainty concerning the three tolerance cases. The larger percentage values in the first row of Table 5.2 indicate that Young's modulus of RVEs is more sensitive to surface distortion than Poisson's ratio. The statement is also illustrated by the large shrinkage of the property boundary along the axis of Young's modulus. The above comparison results and statements can also be proven by comparing standard deviations (σ) between the two properties. For these points selected on and within the boundary, the standard deviations of properties are normalized with their extreme values and summarized in Table 5.3 for the three tolerance cases. From the statistics, the uncertainty in the geometric patterns of the RVEs has a relatively large influence on Young's modulus of the CMMs compared to Poisson's ratio, leading to relatively large percentage differences in Young's modulus.

Table 5.2: Summary of average percentage differences between deterministic and stochastic properties of the seven selected points for the three tolerance cases.

	Case #1 (±10%)	Case #2 (±20%)	Case #3 (±30%)
%diff of E_{11}	0.59%	1.19%	1.79%
%diff of ν_{12}	0.56%	1.14%	1.60%

Table 5.3: Summary of average normalized standard deviations of the seven selected points for the three tolerance cases.

	Case #1 (±10%)	Case #2 (±20%)	Case #3 (±30%)
$\sigma_{normalized}$ of E_{11}	0.0541	0.1082	0.1623
$\sigma_{normalized}$ of ν_{12}	0.0488	0.0964	0.1452

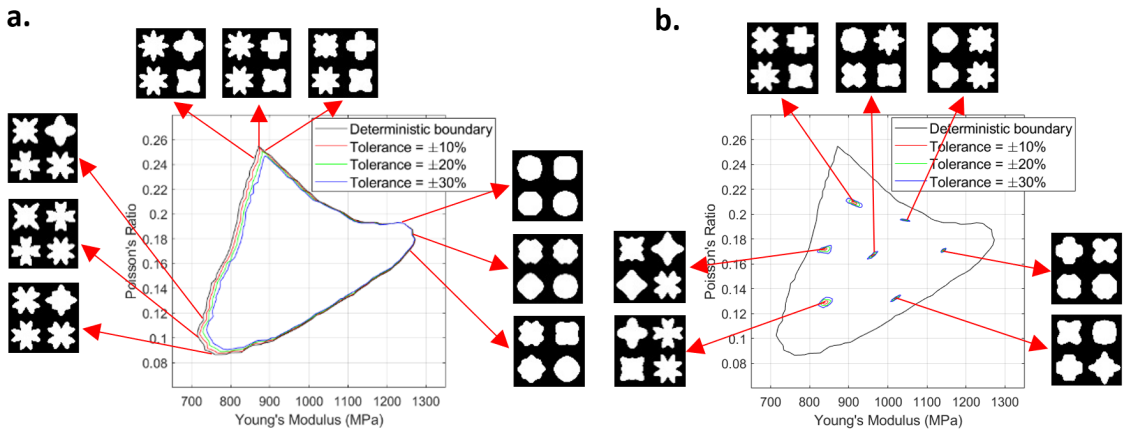


Figure 5.8: Variation intervals of the three tolerance cases for property points selected (a) on the maximal material-property space boundary and (b) within the space boundary.

Chapter 6

GENERATIVE ADVERSARIAL NETWORKS FOR THE INVERSE DESIGN OF SPINODOID METAMATERIALS

Chapter 6 proposes an inverse design framework for spinodoid metamaterials by employing the conditional generative adversarial network (CGAN) that enables the generation of a group of candidate geometric patterns with sets of target effective properties consisting of Young's modulus and shear modulus. Section 6.1 includes the generation of the database and the distribution of the data domain. In Sec. 6.2, this study presents a brief architecture of the CGAN consisting of the discriminator, generator, and solvers. The loss functions of these sub-networks of the CGAN are defined in Sec. 6.2 as well. The detailed architectures of these sub-networks are presented in Sec. 6.3. Section 6.3 also summarizes the performance of the trained CGAN with customized loss functions. Section 6.4 presents the inverse design of spinodoid metamaterials by discussing the reliability and mechanical performance of the generated microstructures.

6.1 Generation of Database

To train the deep learning model, a dataset representing property-geometry relationships is required. The design samples are generated for spinodoid topologies using a random sampling algorithm. Once the geometric patterns of the RVEs are obtained, their effective properties are determined by the homogenization method. In the dataset, each data point consists of a geometric pattern of the RVE and its corresponding property label (Young's modulus and shear modulus). This study converts the geometric pattern into a binary image, represented by a 128×128 element matrix. In the binary image, pixels are Boolean variables taking a value of zero for the void section or one for the solid section. Currently, the dataset contains 51,821 pairs of unique topologies and their corresponding property labels. The data domain is represented by scattering the data points in a 3D property space, with each property label on a particular axis. In Fig. 6.1(a), the ranges for the data domain are 1.01 MPa $< E_{11} <$ 1588.64 MPa, 1.12 MPa $< E_{22} <$ 2264.93 MPa, and 1.03 MPa $<$ $G_{12} <$ 318.75 MPa. The distributions of three properties are illustrated in Fig. 6.1(b). It is noticed that the ranges for Young's modulus values demonstrate a difference. This is because our microstructure generation algorithm has a bias on the spinodal topologies. The bias is mainly induced by the lower bound of design variables, θ_{min}. With the lower bound of the design variables, the generation algorithm is unable to create a geometric pattern that contains all horizontal stripes in the void section while maintaining good structural connectivity of the microstructures. Consequently, the range of E_{11} (Young's modulus in the longitudinal direction) is smaller than that of E_{22} (Young's modulus in the transverse direction). However, by rotating orientations of the microstructures, their property domain can be further expanded, which is demonstrated in the discussion. Besides, since zero values in pixels of binary images might lead to lower activation of the first layer of the network, they are replaced with "-1" in the training process [79]. For consistency with the binary

images, both Young's modulus and shear modulus are normalized such that they range from -1 to 1 based on their extreme values.

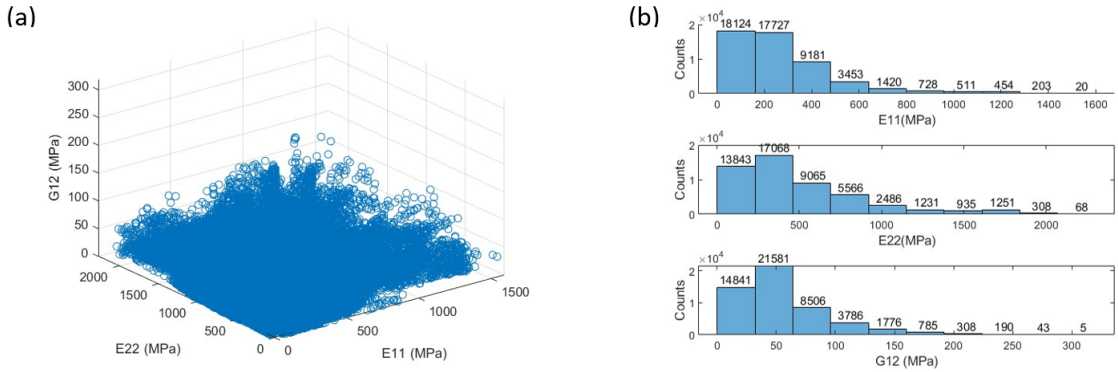


Figure 6.1: Dataset for neural network training: (a) 3D property space of 51,821 randomly created data points and (b) Distributions of the property labels of CGAN for Young's modulus and shear modulus.

6.2 Framework of Inverse Deep Learning Model

For the inverse design, this study aims to devise a CGAN to generate a group of candidate geometric patterns with effective properties that are identical or similar to target properties. The CGAN is a type of deep learning model that achieves data generation by incorporating labels as constraints [61]. Compared to the classic GAN, the CGAN allows the users to generate images of a given type [62]. The architecture of CGAN is comprised of two sub-networks, a generator (G) and a discriminator (D). In the study, the generator takes a random noise vector from the latent space and a target property label as inputs. During the adversarial process, it learns to generate new fake geometric patterns indistinguishable from the training data for specific target properties [63]. Meanwhile, the discriminator learns to distinguish if its input of geometric patterns is real (it belongs to the training data) or fake (produced by the generator) [63]. The two networks are trained concurrently. As the discriminator identifies fake images more precisely, the generator has to generate more

realistic data to fool the discriminator. However, the conventional CGAN might suffer from the overfitting problem as training the discriminator is excessively dependent on the training data [64].

To overcome this problem, an independent solver (S) promotes the generator to yield geometric patterns with objective (target) properties. The solver is a conventional neural network (CNN) applied to bridge the geometric patterns and properties (forward problem). It is a pre-trained network and independent of the two sub-networks. Training the CNN requires minimizing the mean squared error (MSE) objective function, which is defined as minimizing the difference between the actual values and predictions, to solve for the parameters of the CNN, $\omega^{(S)}$. The objective function is defined as:

$$MSE(\omega^{(S)}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{6.1}$$

where y_i is the actual property of spinodoid metamaterials, and $\hat{y}_i$ is the property predicted by the trained CNN. Consequently, the generator's parameters $\omega^{(G)}$ are updated by the discriminator and solver concurrently. The total generator loss is composed of the binary cross-entropy loss (BCE) and mean-squared error loss (MSE) of the generated microstructures. The generator loss L_G is defined as

$$L_G(\omega^{(G)}) = \frac{1}{N}\sum_{i=1}^{N}\{E_z[log(1 - D(G(z|y_i))] + \alpha(y_i - \hat{y}_i)^2\} \tag{6.2}$$

where $\omega^{(G)}$ is the discriminator's parameters, z is the random noise vector, and α is a weighting coefficient that determines the extent to which the loss functions affect the generator. The discriminator loss is composed of the binary cross-entropy loss of the real and generated

microstructures. The discriminator loss L_D is defined as

$$L_D(\omega^{(D)}) = -\frac{1}{N}\sum_{i=1}^{N}\{E_x[logD(x|y_i)] + E_z[log(1 - D(G(z|y_i))]\} \tag{6.3}$$

where $\omega^{(D)}$ is the discriminator's parameters and x shows the real images from the database.

A well-trained generator can create a batch of realistic binary images of spinodal topologies with sets of on-demand elastic moduli consisting of E_{11}, E_{22}, and G_{12}. Figure 6.2 demonstrates the general architecture of the proposed CGAN framework and summarizes the functions of the solver and discriminator. In Fig. 6.2, the dashed and solid lines denote the training process of the generator updated by the discriminator and solver, respectively. In this study, the training of proposed neural networks is implemented in a Google Colab Pro system utilizing NVIDIA V100 GPUs with 32 GB memory.

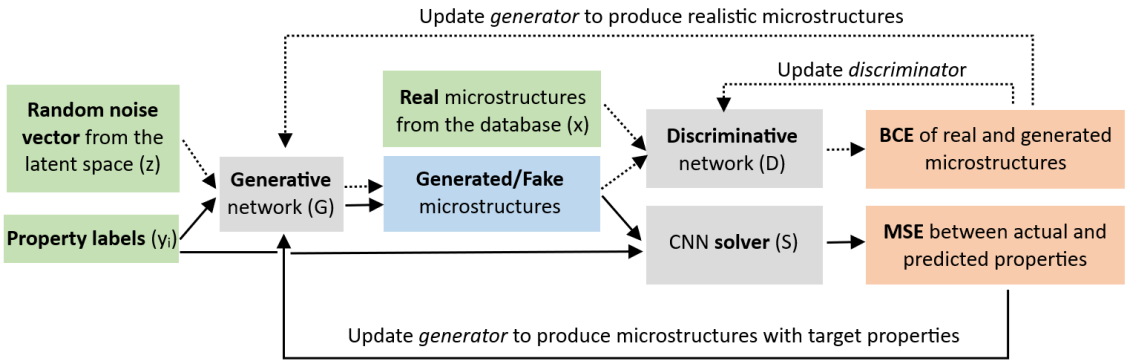


Figure 6.2: General architecture of the proposed CGAN framework. The dashed and solid lines show the training process of the generator updated by the discriminator and solver, respectively.

6.3 Results

6.3.1 Independent convolutional solver

The CNN solver is independent of the generator and discriminator. Therefore, this study first processes the training of the solver with supervised learning to predict the effective properties of a given pattern in the forward direction. In Fig. 6.3, the solver is formed by two parts, the feature maps and the fully connected layers. The first part contains five 2D convolution layers. They extract features from an input binary image and contain 16, 32, 64, 128, and 256 channels. Max-pooling follows each convolution layer to perform down-sampling operations. The output of the feature map is flattened and directly connected with two fully connected layers containing 256 and 128 neurons. Among the 51,821 data points obtained using spinodoid topologies and FEM, 80% are randomly selected as the training dataset, and 20% are used as the test dataset. To minimize overfitting, the solver incorporates dropout layers after each fully connected layer. The early stopping strategy is utilized to reduce model overfitting further. The trainable network parameters are updated by an Adam optimizer with a learning rate of 0.0001. The training process runs for 122

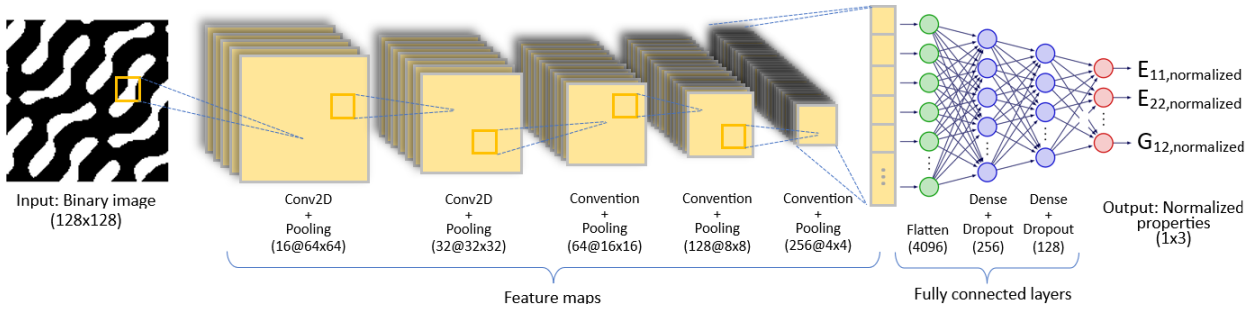


Figure 6.3: Detailed architecture of CNN for the independent solvers.

epochs and the execution time is approximately 1.25 h. As shown in Fig. 6.4(a), the training and validation losses reach minimum values of 9.842e-4 and 9.863e-4, respectively

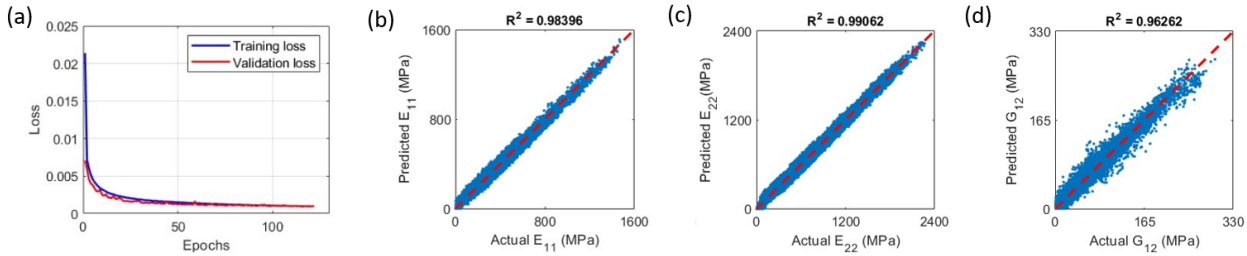


Figure 6.4: Performance of the solver: (a) training history over 122 epochs, (b) (c) and (d) comparison of actual and predicted elastic moduli.

at epoch 122. In addition to the loss, the solver's performance is statistically evaluated by the coefficient of determination (R^2) between the actual and predicted elastic moduli of 10,000 test binary images. As can be observed in Fig. 6.4(b)-(d), the predicted property matches the actual property relatively well for Young's modulus compared to the shear modulus. It is because of their property distribution in the database shown in Fig. 6.1(b). Compared with Young's modulus, the distribution of G_{12} has a relatively large concentration in a low-value range, leading to a relatively small amount of data points in the right tail of the distribution shown in Fig. 6.1(b). It induces bias in property prediction of G_{12} and a reduction in its R^2 value, which is indicated by its wider bandwidth in Fig. 6.4(d) compared with that of Young's modulus in Fig. 6.4(b) and (c). Figure 6.5 summarizes the history of the R^2 of G_{12} over various sizes of the training data points. The figure indicates that the accuracy of the

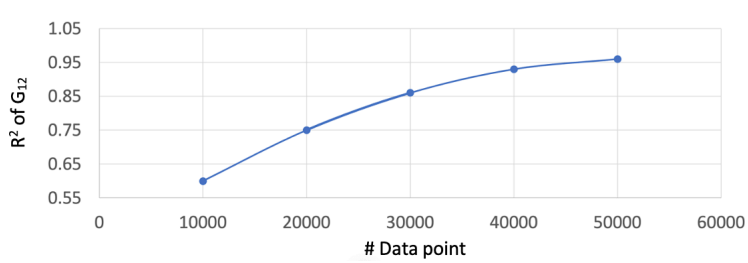


Figure 6.5: History of R^2 value of G_{12} over various data sizes for training the solver.

predicted values of G_{12} can be improved by enlarging the training database. However, as the number of data points increases, the growth rate of R^2 is gradually reduced. It is

caused by the random sampling method for building our database. Due to the central limit theory, the data points tend to follow a Gaussian distribution, which causes them to be centrally located in a portion of the property distribution illustrated in Fig. 6.1. Therefore, continuously enlarging the database is an inefficient approach considering the expensive FE simulations and the gradual decrease in the growth rate of R^2. Inspired by the work of Zheng et. al [80], the CGAN can perform importance sampling at the locations of the data distribution that contains insufficient data points. Thus, the CGAN allows us to generate new microstructures with G_{12} in a large-value range, leading to a reduction in bias in the property prediction of the networks during the training process. If the users are not satisfied with the prediction accuracy of their target properties, they can retrain the networks with a new database enriched by the CGAN. On the whole, however, the high values of R^2 (Fig. 6.4) indicate that our CNN solver has good performance in property predictions for the binary images of spinodoid microstructures.

6.3.2 Conditional generative adversarial network

To solve the many-to-many problem for spinodoid metamaterials, the CGAN is implemented to learn the mapping between the structural topologies and their corresponding properties. Except for the independent solver, our inverse deep learning model is the same as the standard CGAN, which consists of the generator and discriminator. The two neural networks are both realized as CNNs. The generator has two input channels, including 1×256 random noise vectors and a 1×3 property label. They are reshaped and concatenated into a required image format and processed through CNN with upsampling layers, 2D convolution layers, batch normalization, and leaky rectified linear unit (ReLU) activation function [81]. To be compatible with the normalized binary images (Sec. 6.1), the output layer utilizes a Tanh

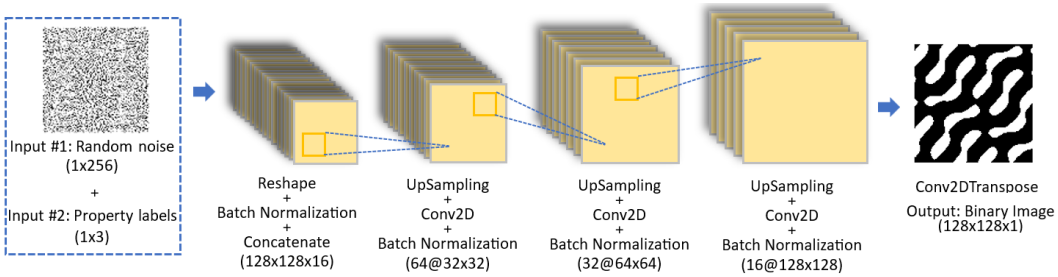


Figure 6.6: Detailed architecture of the generator.

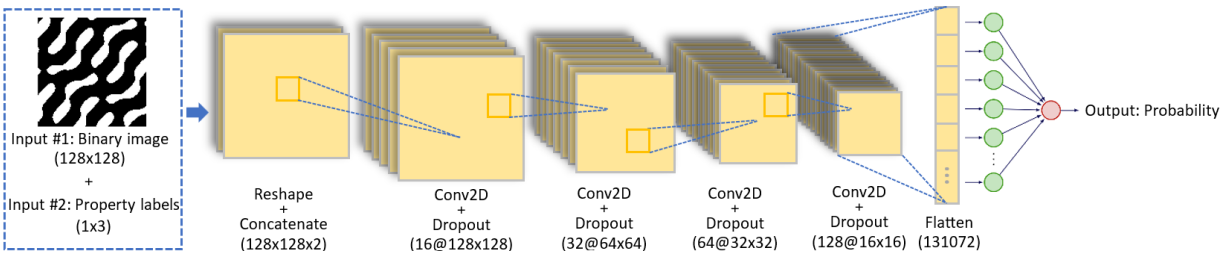


Figure 6.7: Detailed architecture of the discriminator.

activation function to generate new microstructures of spinodoid metamaterials represented by "-1" and "0" values in pixels. For the discriminator, a 1×3 input property label is accompanied by 128×128 input real images in the database or fake images from the generator. Two inputs of the discriminator are reshaped and processed through CNN with 2D convolution layers, a leaky ReLU activation function, and dropout layers. In its output layer, a sigmoid activation function is applied to output values in the range of 0 to 1, which indicates the probability of the sample being from realistic images. The detailed architectures of the generator and the discriminator are shown in Fig. 6.6 and 6.7, respectively.

Our proposed CGAN is trained for 2000 epochs using an Adam optimizer with a learning rate of 0.0004 and a batch size of 32. To reduce property errors between the input labels and generated images, the CGAN in the training process is mainly evaluated in terms of the MSE determined by the solver rather than the classification accuracy retrieved from the discriminator. Figures 6.8(a) and (b) illustrate that the losses of the generator, discriminator, and solver are minimized and stabilized after 1500 epochs of training. This indicates these

networks are fully trained. There are two convergence stages in the MSE history shown in Fig. 6.8(a). Before epoch 70, the discriminator promotes the generator to generate realistic images, resulting in a rapid decrease in the MSE. After epoch 70, the generator is slowly updated by the solver, which progressively reduces the property errors and reaches a minimum MSE of approximately 0.012 after 1500 epochs. The large number of training epochs in the second stage indicates the relatively high difficulty in achieving the low property error compared to generating realistic images. It explains why this study employs the MSE as the major performance indicator of the CGAN. The low value in MSE indicates

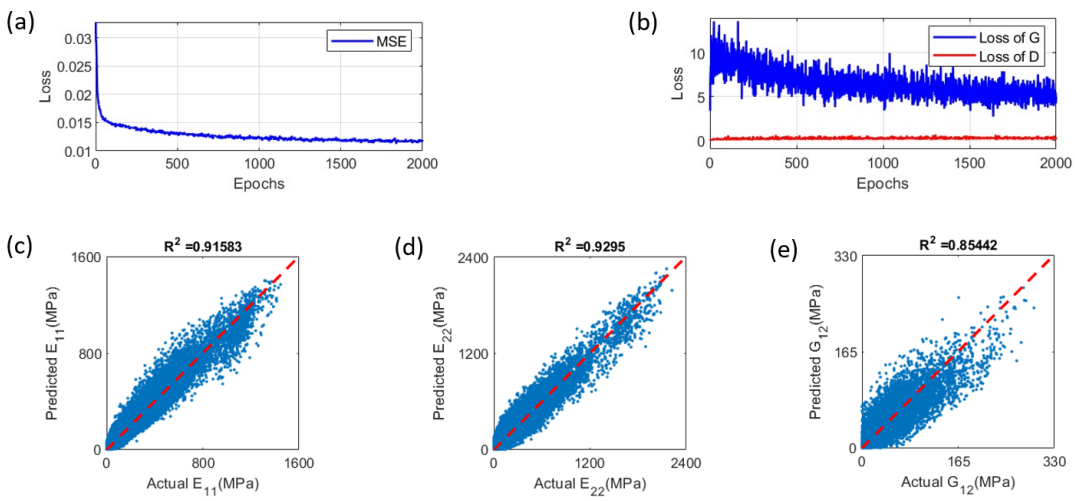


Figure 6.8: Performance of the CGAN: (a) training history of the solver over 2000 epochs, (b) training history of the discriminator and generator over 2000 epochs, (c) (d) and (e) comparison of actual and predicted elastic moduli.

our trained CGAN has the capability to generate a group of realistic microstructures of spinodoid metamaterials with similar or the same user-defined Young's modulus and shear modulus. In addition, the performance of the trained CGAN is also evaluated by comparing each input/actual property and its output/predicted properties of images generated using the CGAN. To plot actual vs predicted values of the elastic moduli, 2,000 property labels are randomly selected in the testing dataset. Our CGAN is used to generate 10 alternative microstructures for each selected label. The plots for the resulting actual vs predicted values

are shown in Fig. 6.8(c)-(e). In Fig. 6.8(c)-(e), the bandwidth of the scatter distribution for each property is wider than the training results of the independent solver (Fig. 6.4(b)-(d)). It is induced by outliers of the alternative microstructures with large property errors. However, in general, the scattered points distribute closely along the bisection line. This represents that our CGAN is capable of generating alternative microstructures with similar or the same properties as the input properties. Among the elastic moduli, the coefficient of determination of G_{12} becomes much smaller compared to the training results of the independent solver. It is due to the error propagation from the solver and the lack of data points in the training dataset of G_{12}. An extensive database allows the CGAN to explore the implicit property-geometry relations more accurately. However, the higher performance of the CGAN requires additional runs of the FE simulation and training process, which are both time-consuming. Thus, by considering the trade-off between computational efficiency and accuracy in our CGAN, the values of the R^2 in Fig. 6.8 are assumed to be acceptable.

6.4 Discussion

6.4.1 Inverse design of spinodoid metamaterials

After assessing the performance of the CGAN, this section utilizes our trained CGAN to perform the inverse design of spinodoid metamaterials. User-defined property labels are the inputs of the CGAN to generate a batch of microstructures with the corresponding elastic moduli. To demonstrate the capacity of our CGAN to enrich data points on the right tail of the histogram in Fig. 6.1(b), property labels are selected and include at least one elastic modulus with a large value. Figure 6.9 shows sample alternative microstructures for each selected property label of [E_{11}, E_{22}, G_{12}], including [1251.4, 302.4, 45.6] MPa, [97.1, 965.7,

25.6] MPa, and [513.2, 342.6, 175.6] MPa. To select an appropriate microstructure among these alternative solutions, the users are required to make a trade-off between accuracy in their corresponding properties and uncertainty in the structural dimension during the

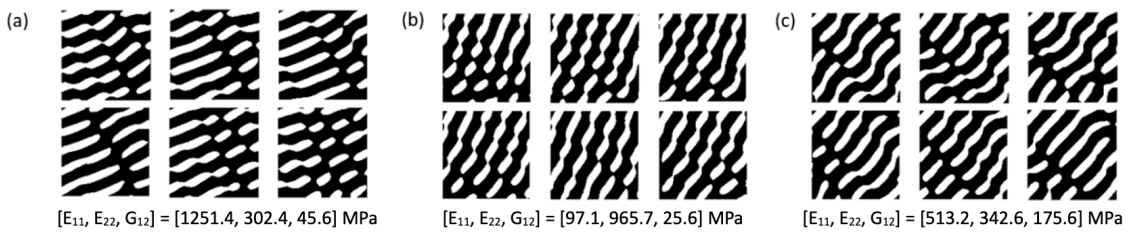


Figure 6.9: Sample alternative microstructures with three different input property labels of the CGAN.

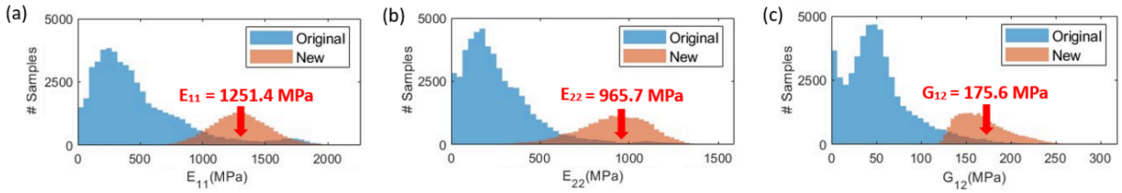


Figure 6.10: Distributions of elastic moduli from the original database and from the newly generated samples with an input condition of (a) $E_{11} = 965.7$ MPa, (b) $E_{22} = 1251.4$ MPa, and (c) $G_{12} = 175.6$ MPa.

additive manufacturing process. Similar to the importance sampling, Fig. 6.10 demonstrates that our CGAN is able to enrich the database in the right tail of its histogram with 20,000 new data points in an efficient way. Compared to the randomly generated method, our proposed inverse design framework does not require spinodoid topologies with geometrical design parameters and property calculations by FEM. Note that the distribution of shear modulus for CGAN-outputted patterns with an input $G_{12} = 150.6$ MPa is left-skewed in Fig. 6.10(c) due to the highly concentrated data points on the low values of the shear modulus in the training database.

To validate the microstructures generated by the CGAN, this study performs FE simulations of the generated microstructures. Table 6.1 summarizes the validation results of sample

microstructures covering wide ranges of corresponding properties. The results include input property labels of the generator, predicted properties of generated microstructures (pre-trained solver), and actual properties of generated microstructures (FE simulation). Since the CGAN employs the MSE of the solvers as part of its loss function, the difference between the input labels and predicted properties is relatively small compared to the actual properties. Owing to the distribution of the training dataset, the difference increases as the values of the elastic moduli increase. In general, the small difference among the validated results for each input label indicates that both the CNN solver and the CGAN have a good performance in mapping property-geometry relations in a forward and inverse fashion, respectively.

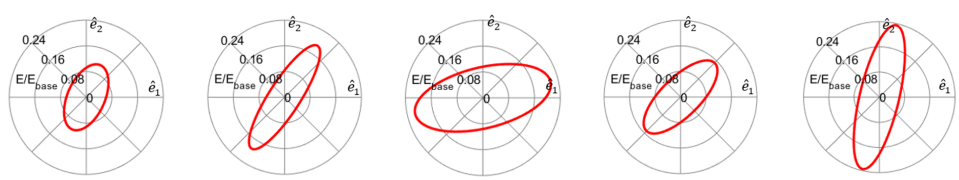
Table 6.1: Validation results of spinodoid microstructures generated by CGAN.

Case #	1	2	3	4	5	6
Spinodoid microstructures generated by CGAN						
E_{11}(input), MPa	1248.60	386.30	98.50	513.60	662.30	199.50
E_{11}(CNN), MPa	1249.50	386.03	98.68	514.17	662.96	199.38
E_{11}(FEM), MPa	1251.06	385.44	98.88	515.30	664.00	199.24
E_{22}(input), MPa	76.20	1911.20	960.30	1225.90	393.40	685.90
E_{22}(CNN), MPa	76.34	1912.66	959.29	1224.99	393.64	686.56
E_{22}(FEM), MPa	76.51	1915.28	957.64	1223.31	393.94	687.74
G_{12}(input), MPa	20.73	50.17	72.08	132.00	160.86	201.62
G_{12}(CNN), MPa	20.58	50.42	72.58	131.08	158.93	204.04
G_{12}(FEM), MPa	20.38	50.77	73.29	129.78	157.19	206.22

6.4.2 Mechanical behavior of spinodoid metamaterials

In our 2D study case, the resulting mechanical behavior of spinodoid metamaterials is represented as a second-order homogenized elastic modulus tensor $\hat{C}_{ij} = [C_{11}, C_{12}, C_{13}; C_{21}, C_{22}, C_{23}; C_{31}, C_{32}, C_{33}]$. The modulus tensor is linearly elastic and obtained by applying average strain $\overline{\varepsilon}_{ij}$ on the RVEs, computing the volume-averaged stress $\overline{\sigma}_{ij}$ by FE simulations, and solving the stress-strain relationship defined as $\overline{\sigma}_{ij} = \hat{C}_{ij}\overline{\varepsilon}_{ij}$. To further expand the design space, the orientation of materials is manipulated to explore structures with outstanding properties. The anisotropic Young's modulus distribution along all directions d in the Cartesian coordinates is computed by the equation defined as $E(d) = (\sum_{i,j=1}^{2} \hat{C}_{ij}^{-1} d_i d_j)^{-1}$ [33]. Table 6.2 summarizes the resulting anisotropic Young's modulus of five representative microstructures. The longitudinal and transverse axes correspond to normalized E_{11} and E_{22}, respectively.

Table 6.2: Anisotropic homogenized Young's modulus of the microstructures generated by CGAN.

Case #	1	2	3	4	5
Spinodoid microstructures generated by CGAN					
Anisotropic properties of Young's modulus					

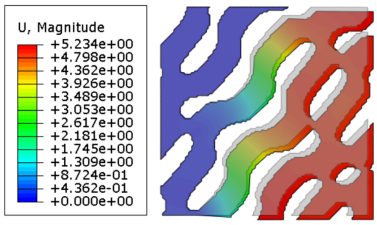
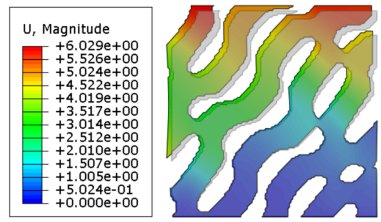
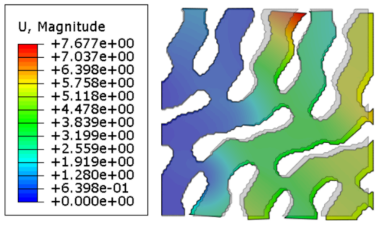
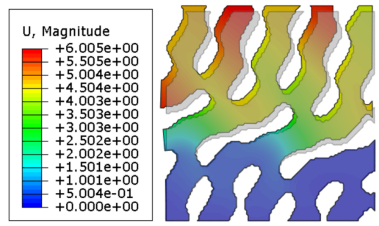
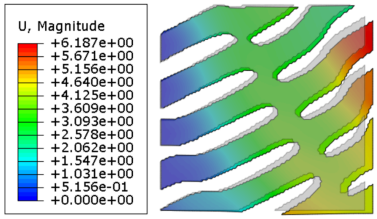
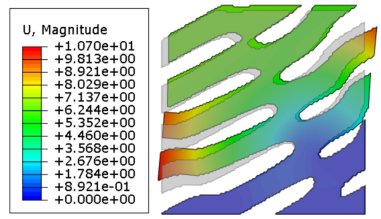
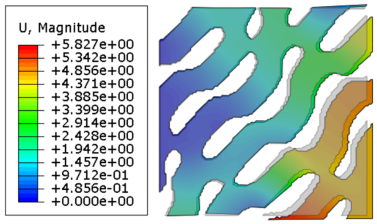
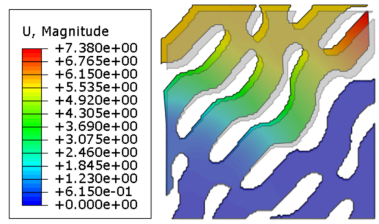
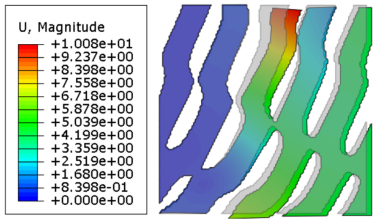
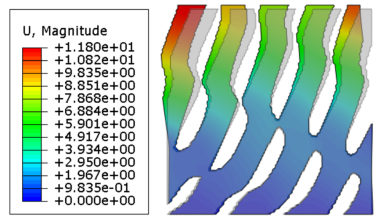
Moreover, this study performs tensile tests and FE simulations to further explore the mechanical behavior of spinodoid metamaterials under a small deformation with a strain ε of 0.03. Table 6.4 summarizes the displacement fields of the selected microstructures under the uniaxial tensile tests in two principal directions. With the displacement fields achieved from ABAQUS, the axial and lateral changes in the displacement of microstructures are calculated

using MATLAB. The Poisson's ratio in Table 6.3 is calculated by the engineering strains, which are derived from the averaged displacements [82]. In Table 6.4, the overall deformation findings prove that our tailored metamaterials can retrieve a negative Poisson's ratio, which is an advantageous feature for aerospace systems, such as deployable and morphing structures.

Table 6.3: Poisson's ratio values of the microstructures shown in Table 6.2.

Case #	1	2	3	4	5
Poisson's ratio, ν_{12}	-0.3010	0.0346	0.2551	-0.0953	0.2155
Poisson's ratio, ν_{21}	-0.0908	0.0288	0.3682	-0.0875	0.1254

Table 6.4: Displacement fields of the microstructures generated by CGAN.

Case #	Spinodoid microstructures generated by CGAN	**Displacement field of uniaxial tensile test**	
		Longitudinal direction (X)	Transverse direction (Y)
1			
2			
3			
4			
5			

Chapter 7

CONCURRENT MULTI-SCALE TOPOLOGY OPTIMIZATION OF POROUS METAMATERIALS

Chapter 7 presents a multi-scale design framework for the concurrent topology optimization of 2D structures with porous metamaterials. The objective of this study is to minimize the compliance of the structure by optimizing the material distributions at the micro and macro scales. The optimization problem is formulated in Sec. 7.1 with the constraint of the total amount of materials. A quadratic programming method is applied to solve this optimization problem, with its mathematical formulation demonstrated in Sec. 7.1. Section 7.2 summarizes the optimization results in two design scenarios: for a cantilever beam and a Messerschmitt–Bölkow–Blohm (MBB) beam. What's more, Section 7.2 discusses whether the multi-scale structures with porous metamaterials can increase the performance of the structures with solid materials, and the effects of structural dimensions (macro-scale: mesh size; micro-scale: pixel dimension) on accuracy and efficiency in the sensitivity analysis.

7.1 Optimization Problem

7.1.1 Optimization formulation

The multi-scale optimization problem aims to optimize material distributions in both macro and micro scales to retrieve minimum compliance of the structure while defining a constraint on the volume fraction of the material. At the macro-scale, the design domain is assumed to be the rectangular beam shown in Fig. 7.1. To perform topology optimization (TO), the design domain is discretized by quadrilateral finite elements. The aspect ratio of the macrostructure is defined by the number of elements in the longitudinal ($nelx$) and transverse ($nely$) directions of the design domain. Following the work of Bendsøe [83] and Zhou et al. [84], this study employs a density-based approach to approximate Young's modulus of the elements, E_e. The approach is called the Solid Isotropic Material with Penalization (SIMP) method and defines E_e as:

$$E_{i,e} = E_{min} + (x_{i,e})^p (E_0 - E_{min}) \tag{7.1}$$

where E_{min} is extremely small (numerically negligible) stiffness assigned to void sections in the design domain, E_0 is the stiffness of the base material, $x_{i,e}$ is the density assigned to the element i, and p is a penalty factor to ensure the black-and-white solutions of the TO. This study investigates two design scenarios corresponding to two external boundary conditions shown in Fig. 7.1: the cantilever beam and the MBB beam. At the micro-scale, each discretized element has an identical microstructure, which is the geometric pattern of RVEs of porous metamaterials (Fig. 2.1). Therefore, the microstructure in the design domain is directly controlled by two geometric variables ζ_1 and ζ_2 in Eq. 2.1. The energy-based homogenization in Sec. 2.2.1 is applied to determine the homogenized elastic modulus

tensor ($\hat{C}_e$) of the microstructures:

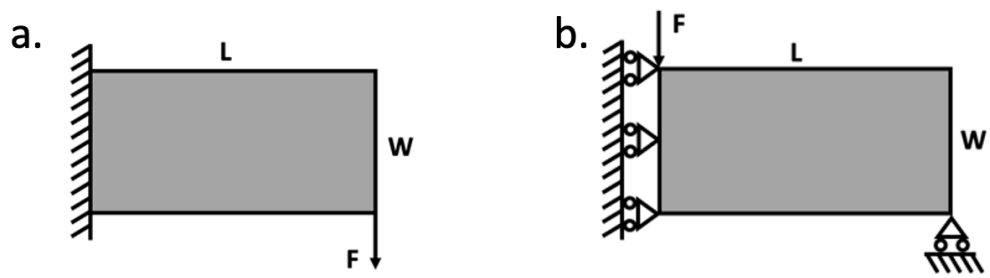


Figure 7.1: The design domains and boundary conditions for the topology optimization of (a) a cantilever beam and (b) a symmetric MBB beam.

$$\hat{C}_e = \begin{bmatrix} C_1 & C_2 & 0 \\ C_3 & C_4 & 0 \\ 0 & 0 & C_5 \end{bmatrix} \tag{7.2}$$

where $C_1 = C_4$ and $C_2 = C_3$ since the geometric pattern of porous metamaterials is four-fold symmetric, leading to isotropic homogenized properties. Consequently, the mathematical formulation of the multi-scale TO problem to minimize compliance (ϕ) can be written as:

$$\begin{aligned} &\text{Minimize} : \phi(x, \zeta_1, \zeta_2) = U^T K U = \textstyle\sum_{i=1}^{N} [E_{min} + (x_{i,e})^p (E_0 - E_{min})][(u_{i,e})^T k_{i,e} u_{i,e}] \\ &\text{Find} : x, \zeta_1, \zeta_2 \\ &\text{Subject to} : \textstyle\sum_{i=1}^{N} x_{i,e} = f_v N \\ &\qquad\qquad KU = F \\ &\qquad\qquad 0 \leq x \leq 1,\ -0.15 \leq \zeta \leq 0.15 \end{aligned} \tag{7.3}$$

where ϕ is the compliance, U is the global displacement vector of the beam, K is the global stiffness matrix of the beam, N is the number of discretized elements in the design domain, u is the local displacement vector of the element, f_v is the prescribed volume fraction of the

material, and F is the external force applied on the beam. The stiffness of the discretized element ($k_{i,e}$) is determined by:

$$\int_{\Omega} (B_e)^T \hat{C}_e(\zeta_1, \zeta_2) B_e d\Omega \tag{7.4}$$

where B_e is the B matrix for the quadrilateral element and $\hat{C}_e$ is the homogenized elastic modulus tensor of porous metamaterials in Eq. 7.2.

7.1.2 Optimization method

The optimization problem is classified as a non-linear programming (NLP) problem due to its objective function containing a highly non-linear term, which is the homogenized elastic modulus tensor. This study solves the optimization problem using sequential quadratic programming (SQP) in MATLAB. The sub-quadratic problem in the iterative process of the SQP is defined with a quadratic objective function and a linear constraint as:

$$\begin{aligned} &\bigtriangledown^2 \phi^{(m)} d + A^T \lambda = - \bigtriangledown \phi^{(m)} \\ &Ad = 0 \end{aligned} \tag{7.5}$$

where m is the number of iterations in the SQP, A is the matrix of active constraints on the volume fraction of material, λ is the Lagrange multiplier of the Hessian matrix $\bigtriangledown^2 \phi$, and d is the search direction of the design variables. In the optimization process, the SQP approach successively determines the search direction d in the sub-QP problem and then updates the design variables using a line search procedure shown in Sec. 2.3.1. Since the homogenized elastic modulus tensor ($\hat{C}_e$) is implicit, its derivatives with respect to the geometric variables are approximated by the central difference method in the sensitivity analysis:

$$
\begin{aligned}
\frac{\partial \hat{C}_e}{\partial \zeta_1} &\approx \frac{\hat{C}_e(\zeta_1+\Delta,\zeta_2)-\hat{C}_e(\zeta_1-\Delta,\zeta_2)}{2\Delta} \\
\frac{\partial \hat{C}_e}{\partial \zeta_2} &\approx \frac{\hat{C}_e(\zeta_1,\zeta_2+\Delta)-\hat{C}_e(\zeta_1,\zeta_2-\Delta)}{2\Delta} \\
\frac{\partial^2 \hat{C}_e}{\partial \zeta_1^2} &\approx \frac{\hat{C}_e(\zeta_1+\Delta,\zeta_2)-2k_e(\zeta_1,\zeta_2)+\hat{C}_e(\zeta_1-\Delta,\zeta_2)}{\Delta^2} \\
\frac{\partial^2 \hat{C}_e}{\partial \zeta_2^2} &\approx \frac{\hat{C}_e(\zeta_1,\zeta_2+\Delta)-2\hat{C}_e(\zeta_1,\zeta_2)+\hat{C}_e(\zeta_1,\zeta_2-\Delta)}{\Delta^2} \\
\frac{\partial^2 \hat{C}_e}{\partial \zeta_1 \partial \zeta_2} &\approx \frac{\hat{C}_e(\zeta_1+\Delta,\zeta_2+\Delta)-\hat{C}_e(\zeta_1+\Delta,\zeta_2-\Delta)-\hat{C}_e(\zeta_1-\Delta,\zeta_2+\Delta)+\hat{C}_e(\zeta_1-\Delta,\zeta_2-\Delta)}{4\Delta^2}
\end{aligned}
\tag{7.6}
$$

where Δ is the step size. The impact of Δ on the tensor ($\hat{C}_e$) is shown in Appendix A. With a MATLAB optimization toolbox called *fmincon*, the Hessian matrix $\bigtriangledown^2\phi$ can be directly approximated by the means of a Quasi-Newtown method called Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [42]. However, the parameter space of the porous metamaterials is complex [44]. A small variation in geometric variables ζ_1 and ζ_2 might lead to large variations in values of the homogenized elastic modulus tensor. To ensure the accuracy and stability of the sensitivity analysis, the first and second derivative of the objective function is numerically calculated by Hooke's law and partial differential equations in each sub-quadratic problem of the SQP [42]. The first derivatives of the objective function are shown in Eq. 7.7. The second derivatives of the objective function are shown in Eqs. 7.8, 7.9, 7.10, and 7.11. Consequently, the closed-form expression of the first and second derivatives of the objective function can be summarized and expressed by Eqs. 7.13 and 7.14, respectively. Besides, this study imposes the mesh-dependency filter to ensure the existence of realistic solutions and avoid numerical instabilities of the macrostructures, such as checkerboard patterns (Fig. 7.2) [40]. The filter modifies the sensitivities $\frac{\partial \phi}{\partial x}$ as in Eq. 7.15. The sensitivities associated with the geometric variables are modified similarly as in Eq. 7.15 [85].

$$\begin{aligned}
\frac{\partial \phi}{\partial x} &= -\sum_{i=1}^{N}[p(x_{i,e})^{p-1}(E_0 - E_{min})][(u_{i,e})^T k_{i,e} u_{i,e}] \\
\frac{\partial \phi}{\partial \zeta_1} &= -\sum_{i=1}^{N}[E_{min} + (x_{i,e})^p (E_0 - E_{min})][(u_{i,e})^T \frac{\partial k_{i,e}}{\partial \zeta_1} u_{i,e}] \\
\frac{\partial \phi}{\partial \zeta_2} &= -\sum_{i=1}^{N}[E_{min} + (x_{i,e})^p (E_0 - E_{min})][(u_{i,e})^T \frac{\partial k_{i,e}}{\partial \zeta_2} u_{i,e}]
\end{aligned} \tag{7.7}$$

$$\frac{\partial^2 \phi}{\partial x_{i,m} \partial x_{i,n}} = \begin{cases} 0, & m \neq n, \\ \sum_{i=1}^{N} 2[p(x_{i,e})^{p-1}(E_0 - E_{min})]^2 & \\ [E_{min} + (x_{i,e})^p (E_0 - E_{min})]^{-1}[(u_{i,e})^T k_{i,e} u_{i,e}], & m = n. \end{cases} \tag{7.8}$$

$$\frac{\partial^2 \phi}{\partial x_{i,m} \partial \zeta_{1,n}} = \begin{cases} 0, & m \neq n, \\ \sum_{i=1}^{N}[p(x_{i,e})^{p-1}(E_0 - E_{min})][(u_{i,e})^T \frac{\partial k_{i,e}}{\partial \zeta_1} u_{i,e}], & m = n. \end{cases} \tag{7.9}$$

$$\frac{\partial^2 \phi}{\partial x_{i,m} \partial \zeta_{2,n}} = \begin{cases} 0, & m \neq n, \\ \sum_{i=1}^{N}[p(x_{i,e})^{p-1}(E_0 - E_{min})][(u_{i,e})^T \frac{\partial k_{i,e}}{\partial \zeta_2} u_{i,e}], & m = n. \end{cases} \tag{7.10}$$

$$\frac{\partial^2 \phi}{\partial \zeta_{1,m} \partial \zeta_{1,n}} = \begin{cases} 0, & m \neq n, \\ \sum_{i=1}^{N}[E_{min} + (x_{i,e})^p (E_0 - E_{min})] & \\ \{(u_{i,e})^T [2(\frac{\partial k_{i,e}}{\partial \zeta_1})^2 k_{i,e}^{-1} - \frac{\partial^2 k_{i,e}}{\partial \zeta_1^2}] u_{i,e}\} & m = n. \end{cases} \tag{7.11}$$

$$\frac{\partial^2 \phi}{\partial \zeta_{1,m} \partial \zeta_{2,n}} = \begin{cases} 0, & m \neq n, \\ \sum_{i=1}^{N}[E_{min} + (x_{i,e})^p (E_0 - E_{min})] & \\ \{(u_{i,e})^T [2\frac{\partial k_{i,e}}{\partial \zeta_1}\frac{\partial k_{i,e}}{\partial \zeta_2} k_{i,e}^{-1} - \frac{\partial^2 k_{i,e}}{\partial \zeta_1 \partial \zeta_2}] u_{i,e}\} & m = n. \end{cases} \tag{7.12}$$

$$\nabla \phi = \begin{bmatrix} \frac{\partial \phi}{\partial x_1} & \frac{\partial \phi}{\partial x_2} & \dots & \frac{\partial \phi}{\partial x_N} & \frac{\partial \phi}{\partial \zeta_1} & \frac{\partial \phi}{\partial \zeta_2} \end{bmatrix}^T \tag{7.13}$$

$$\nabla^2\phi = \begin{bmatrix} \frac{\partial^2\phi}{\partial x_1^2} & 0 & 0 & 0 & \frac{\partial^2\phi}{\partial x_1\partial\zeta_1} & \frac{\partial^2\phi}{\partial x_1\partial\zeta_2} \\ 0 & \frac{\partial^2\phi}{\partial x_2^2} & 0 & 0 & \frac{\partial^2\phi}{\partial x_2\partial\zeta_1} & \frac{\partial^2\phi}{\partial x_2\partial\zeta_2} \\ 0 & 0 & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{\partial^2\phi}{\partial x_N^2} & \frac{\partial^2\phi}{\partial x_N\partial\zeta_1} & \frac{\partial^2\phi}{\partial x_N\partial\zeta_2} \\ \frac{\partial^2\phi}{\partial x_1\partial\zeta_1} & \frac{\partial^2\phi}{\partial x_2\partial\zeta_1} & \cdots & \frac{\partial^2\phi}{\partial x_N\partial\zeta_1} & \frac{\partial^2\phi}{\partial\zeta_1^2} & \frac{\partial^2\phi}{\partial\zeta_1\partial\zeta_2} \\ \frac{\partial^2\phi}{\partial x_1\partial\zeta_2} & \frac{\partial^2\phi}{\partial x_2\partial\zeta_2} & \cdots & \frac{\partial^2\phi}{\partial x_N\partial\zeta_2} & \frac{\partial^2\phi}{\partial\zeta_1\partial\zeta_2} & \frac{\partial^2\phi}{\partial\zeta_2^2} \end{bmatrix} \tag{7.14}$$

Figure 7.2: Checkerboard pattern achieved in the optimization process without applying the sensitivity filter.

$$\frac{\partial\hat{\phi}}{\partial x} = \frac{1}{x_e\sum_{f=1}^{N}\hat{H}_f}\sum_{f=1}^{N}(\hat{H}_f x_f)\frac{\partial\phi}{\partial x_f}, \tag{7.15}$$

where $\hat{H}_f$ is a weight factor and written as $\hat{H}_j = r_{min} - dist(e, f)$. r_{min} denotes as the filter radius, and $dist(e, f)$ denotes the distance between the centers of elements e and f [40].

7.2 Results and Discussion

7.2.1 Optimization results

In this study, the proposed concurrent multi-scale TO framework is used to optimize the cantilever beam (Fig. 7.1a) and the MBB beam (Fig. 7.1b). In each design scenario, the framework is performed under three study cases with different mesh sizes of the microstruc-

tures, consisting of 40×20, 60×30, and 80×40. The aspect ratio ($nelx : nely$) of the design domain is maintained the same and is equal to $2 : 1$. This study represents the microstructures of porous metamaterials as binary images with the dimensions of 128×128. The design domain is filled with a homogeneous and isotropic base material with Young's modulus E_0 and $\nu_0 = 0.3$. In the constraint, the volume fraction f_v of the material is set to 50%, which means that the total amount of material in two scales is equal to 50% of the solid beam. The penalty factor in the SIMP approach is set to 4 for all the design scenarios. To avoid the checkerboard patterns, the study solves the optimization problem by using sensitivity filtering with r_{min} that equals to 0.03 times the width of the design domain ($nelx$), i.e. r_{min} = 1.2, 1.8, and 2.4 for different mesh sizes. For simplicity, this study defines Young's modulus of the base material $E_0 = 1$ and normalizes both length and width in the design domain to avoid unit dependency in the following design scenarios. As an initial guess, the density values x_e of all discretized elements are set to 0.5, and the geometry variables $[\zeta_1, \zeta_2]$ are set to $[-0.15, -0.15]$.

The optimal designs of the cantilever beam and the MBB beam with isotropic porous architectures are shown in Fig. 7.3 and 7.4, respectively. At the macro scale, the optimization results show that the edges between black-and-white solutions get smoother as the mesh size increases. Moreover, in the same design scenario (boundary conditions), the minimum compliance values ϕ increase as the mesh size of the study case increases. This can be explained by the compliance's converging behaviors with the mesh size. Under the same boundary condition and aspect ratio, the compliance values continue to increase as the mesh size increases until it reaches its converging point (discussed in Sec. 7.2.3). It also explains the increased stability of the optimized macrostructures in the design scenarios of the MBB beam shown in Fig. 7.4. As the mesh size increases, the optimized MBB structures become more and more similar to each other. Besides, the optimized macrostructures of the MBB beam are in

agreement with the literature [40, 41, 86], which validates the reliability of our proposed TO framework. At the micro-scale, the optimization results show that the optimal designs of microstructures vary in different design scenarios but have similar geometric variable values that approach zero. The noticeable variations in the geometric patterns of the optimized microstructures can be explained by a high sensitivity of the design variables and their corresponding geometric patterns. The high sensitivity might induce highly fluctuating values to the derivatives of the homogenized tensor when slightly modifying their corresponding design variables. The instability in the sensitivity analysis can be overcome by increasing the dimensions of the binary images of microstructures (discussed in Sec. 7.2.3). Apart from that, the increases in the mesh size achieve stability in the optimal design. In Fig. 7.4, the optimal solution of the MBB beam is similar when $nelx \geq 60$. In Fig. 7.5, optimizing the density values of the macrostructure leads to a dramatic reduction in objective values. In the next few iterations, the macrostructure becomes stable, and then the SQP focuses on optimizing the microstructure by continuously updating the geometric variables. The objective value smoothly converges to its optimal in the last few iterations, which indicates the reliability of the proposed framework.

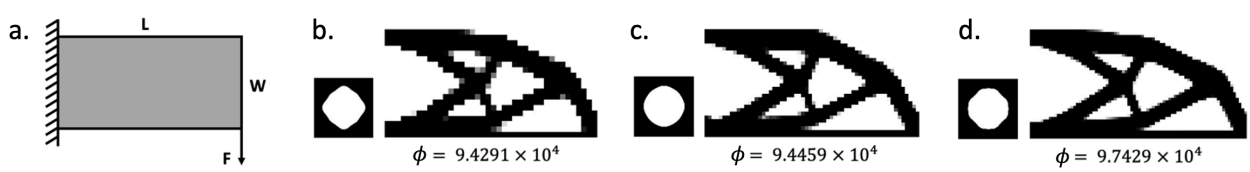


Figure 7.3: Optimization results of (a) the cantilever beam with (b) 40×20, (c) 60×30, and (d) 80×40 elements in its macrostructure.

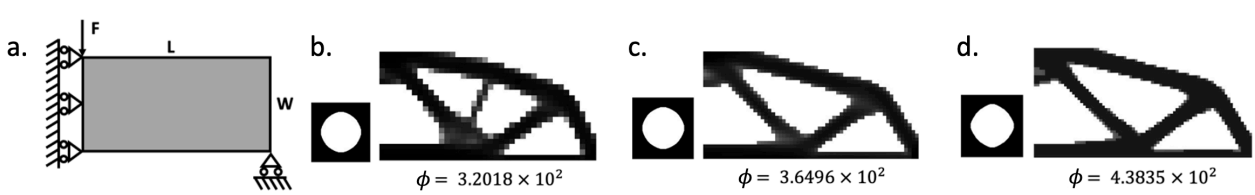


Figure 7.4: Optimization results of (a) the MBB beam with (b) 40×20, (c) 60×30, and (d) 80×40 elements in its macrostructure.

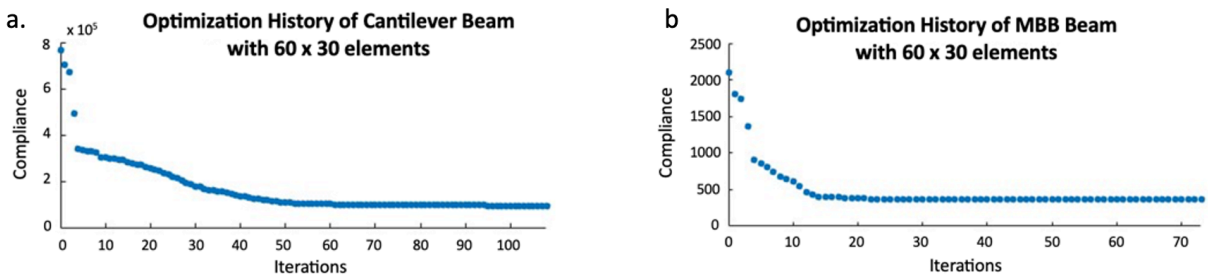


Figure 7.5: Optimization history of (a) the cantilever beam and (b) the MBB bean with 60×30 elements.

7.2.2 Comparison of the optimal compliance for the CMMs and solid materials

This section investigates the benefit of replacing solid materials with porous metamaterials in the beam structures. This is studied by comparing the optimal results between the experimental group (the beam structures with porous metamaterials) and the control group (the beam structures with solid materials). In the comparison, the experimental group is the multi-scale TO of the beam under two different boundary conditions shown in Fig. 7.1. Its optimal results are illustrated in Fig. 7.3 and 7.4. The control group has the optimal results obtained from the classical SIMP method. Instead of the multi-scale structure, this group only focuses on optimizing the beam structure at the macro-scale with the settings of the same base material, mesh elements, radius of the filter, and the penalization factor as the experimental group. In other words, the microstructures in the control group are not considered as the design space and are constantly filled with solid materials during the optimization process.

The comparison results of the two groups with a mesh size of 80×40 are shown in Fig. 7.6 (cantilever beam) and 7.7 (MBB beam). In the two study cases, the optimized compliance in the control group is always larger than the experimental group. It indicates that the

replacement with identical porous metamaterials is unable to improve the stiffness of the beam under these two boundary conditions with the constraint on the volume fraction of materials. The highly compliant behavior of the experimental group is induced by the porosity of the microstructures, which is equal to 0.35. The violated designs of the multi-scale structure might be induced by the limitation of design freedom, in which the identical unit cells are periodically distributed in the design domain. To overcome the issue, the unit cells could have various geometry patterns and be non-periodically distributed in the domain. In short, the underlying materials of the multi-scale structure become heterogeneous. In that case, the microstructures require different combinations of geometric variables $[\zeta_1, \zeta_2]$ to define different patterns of their unit cells. Furthermore, the design freedom can be increased by selecting anisotropic metamaterials, such as spinodoid metamaterials. Controlling the orientations of these materials can contribute to a comprehensive and seamless range of anisotropic mechanical properties, resulting in an enlarged design space [33, 85].

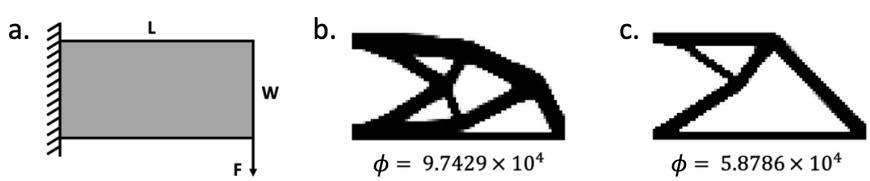


Figure 7.6: Comparison results for (a) the cantilever beam with 80×40 elements containing (b) metamaterials (experimental group) and (c) solid materials (control group).

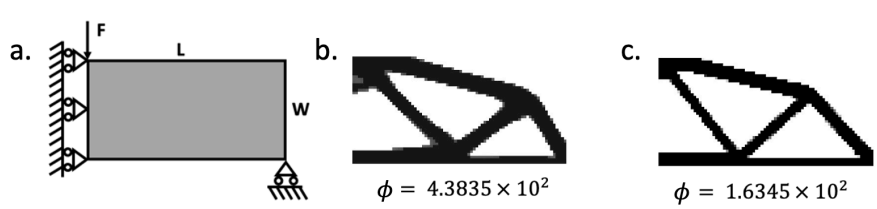


Figure 7.7: Comparison results for (a) the MBB beam with 80×40 elements containing (b) metamaterials (experimental group) and (c) solid materials (control group).

7.2.3 Convergence in sensitivity analysis

In sensitivity analysis, the local truncation error of the sensitivities depends on the selection of the finite differencing (FD) method and step sizes. Since the central differencing approximation is more accurate than the forward and backward differencing approximations [87], it is selected as the FD method of this study to calculate the sensitivities to the geometric variables. To determine an approximate step size, this study plots convergence histories of the sensitivities associated with the step size as shown in Fig. A.1. Figure A.1 shows that the sensitivities vary dramatically when the step size is approximately smaller than 0.02. However, the step size that is too large may create instabilities and affect the reliability in FD approximation [88]. Therefore, the step size is set to 0.02 in this study.

In addition to the two factors mentioned above, the dimensions and mesh sizes of the microstructures can also affect the error in FD approximation. Figure 7.8 illustrates the convergence histories of the homogenized stiffness k that varies with the dimension at the micro-scale. Figure 7.9 illustrates the convergence histories of the compliance ϕ that varies with the mesh size at the macro-scale. From Fig. 7.8 and 7.9, the data stability and simulation duration increase significantly with large values in both the dimension and mesh size. In this case, it is necessary to find a reasonable balance between data quality and simulation duration for practical usage. Large values in the dimension and mesh size help alleviate instabilities in property determination and increase accuracy in the FD approximation. However, it reduces the simulation speed in the optimization process.

To increase the speed while maintaining stability, the automatic differentiation can be applied in future studies by using DNNs (deep neural networks). The DNN-based surrogate model accelerates the property determination by bypassing the simulation and beneficially provides the sensitivities of its input design variables, which can be easily computed by

back-propagating the gradients from the stiffness (outputs of DNNs) to the design variables (inputs of DNNs) via the chain rule for differentiation [85].

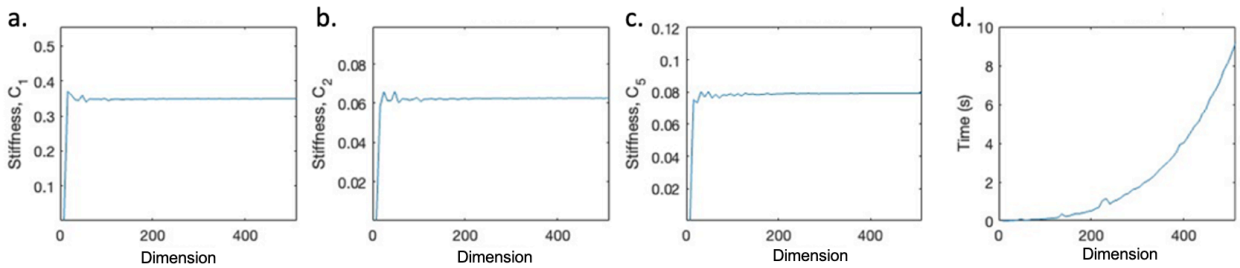


Figure 7.8: Convergence history of the homogenized stiffness of microstructures: (a) C_1, (b) C_2, and (c) C_5; (d) run time of the stiffness calculation associated with various dimensions of the microstructure.

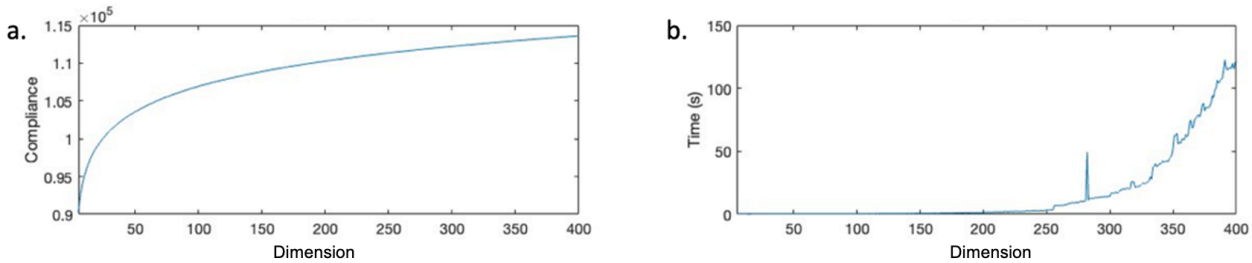


Figure 7.9: Convergence history of (a) the compliance of macrostructures; (b) run time of the compliance calculation associated with various dimensions of the macrostructure.

Chapter 8

CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

This study proposes an experience-free and systemic methodology for exploring the parameter space of cellular mechanical metamaterials using the micro-genetic algorithm with a new search strategy. It opens new avenues to harness genetic algorithms to realize user-defined properties. This method adopts the representative volume element homogenization method for the multi-scale material system to predict its effective properties in ABAQUS (Sec. 2.2.2). The homogenization method is not only compatible with isotropic materials but also with anisotropic composite materials. The broad application of the homogenization method provides possible future research as an extension of the current study. In addition, this study evolves the geometric patterns of RVEs using the new searching strategy based on the micro-generic algorithm (μGA). The experimental results in Sec. 3.3 illustrate that the strategy avoids premature convergence in solving an inverse design problem using μGA, and maintains high accuracy in optimal results while requiring no prior information about the pattern-property pairs. In Sec. 3.4, compared to the standard exhaustive genetic algorithm (SGA), it achieves relatively low computational expense with the small population size and the conditionally reduced parameter space. Moreover, this study examines four different ran-

dom point generators in the standard genetic algorithm. In a high-dimensional parameter space (Sec. 3.3.3), the small population size minimizes the impact on the objective function values with any alternative generators mentioned in this study.

A comprehensive material-property space of cellular mechanical metamaterials can be achieved by an exploration approach proposed in this study and facilitated by the selection algorithm of objective properties and sampling algorithm of property points. The 2D material-property space resulting from the proposed approach is a reliable representation of true combinations of two properties under consideration in academic and industrial applications. The discussion section of this part of the study (Sec. 4.3.1) demonstrates the use of material property spaces of multiple families of the CMMs as a material selection chart to effectively screen out the inappropriate families based on their space region and intuitively pick geometric patterns that are easy to manufacture from the overlapping areas. In addition, this study (Sec. 4.3.2) compares the performance of our proposed sampling algorithm to traditional algorithms for building a database of the NN and finds that our algorithm obtained broader and more evenly distributed property points than traditional algorithms for the same number of samples in the data domain, which guarantees the well-applicability of the NN in the material design problem. As a practical use case, our study considers one of the defects in additive manufacturing, called surface distortion in FFF printing. Uncertainty in the material-property space (combinations of two properties) associated with this defect can be visually demonstrated as variation intervals of its space boundary and numerically represented as variations in its space area under the different tolerance cases (Sec. 5.2).

Furthermore, this study also addresses the many-to-many inverse problem of spinodoid metamaterials by introducing a design framework based on CGANs (Sec. 6.2). The developed CGAN framework leverages a pre-trained independent solver (Sec. 6.3.1) to facilitate our deep-learning network and achieve accurate predictions for highly nonlinear property-

geometry relations. To address the inverse design problem, the proposed framework realizes microstructures with user-defined Young's modulus and shear modulus values. Once it is well-trained, the deep-learning network can bypass numerical simulations and accelerate the tailoring of microstructures to achieve various desired properties. With our CGAN framework, users can obtain a batch of optimal designs targeting elastic moduli in a highly efficient way. These designs (Sec. 6.4.1) are alternative solutions to the many-to-many problems and promote high freedom in the fabrication process. The results of our framework for mechanical behavior confirm that the presented 2D spinodoid metamaterials have superior features, including negative Poisson's ratio and an extensive design space with anisotropic properties.

Additionally, this study develops a concurrent multi-scale topology optimization (TO) framework with isotropic porous metamaterials based on the works of Liu et al. [42] and Gao et al. [43]. This study defines a TO problem that aims to minimize compliance of the engineering beam by optimizing structures in both micro- and macro-scale with the constraints on the total amount of material in the design domain. In the multi-scale structure, the mechanical behaviors in two scales are approximated by using density-based methods, including the SIMP approach (macro-scale) and the EBHM method (micro-scale). This study solves the TO problem by using a gradient-based optimization algorithm called the SQP. The sensitivity analysis in the SQP is facilitated by the central difference approximation and the analytical derivation of the first and second derivatives of the objective function. The application of the developed framework is demonstrated through two study cases: the cantilever beam and the MBB beam with different mesh sizes. Although this study is unable to improve the stiffness of the beam by replacing solid structures with equivalent porous metamaterials in total volume fraction, the presented framework is sufficient to extend to other metamaterials with heterogeneous unit cells distributed in the design domain. The extended framework is anticipated to potentially obtain optimal designs with improved mechanical

performance owing to their strong anisotropy and diverse geometric features.

8.2 Future Works

In this section, future works will be discussed in terms of the materials and methodologies. For the materials, future studies can be extended to 3D, other types of metamaterials, and non-linear properties. For the methodologies, some drawbacks and extensions of the proposed design framework are worth examining in the future.

1) Inverse design framework with the GA: Owing to the GA's capability of imitating the natural evolution, the proposed inverse design method of porous metamaterials with GA can easily be applied to the inverse design of other cellular metamaterials with high dimensional discrete design parameters, such as truss-like structures and other cellular metamaterials where geometric patterns of RVEs are defined by an ensemble of binary values (genes).

2) Material property space exploration: First, it is not difficult to increase objective properties selected on the space boundary to explore more undiscovered property points that are possibly exterior to the current space. It could lead to a reduction in the number of space expansion loops. However, it is challenging to determine the appropriate amount of possible property points to finalize a material-property space since this requires balancing the benefits of comprehensive solution distributions in the space against the drawbacks of high computational expenses of the GA. The effectiveness of the number and diversity of possible solutions in GA is an essential question for future studies. Secondly, apart from uniform sampling, other random sampling methods are available for sampling objective properties to determine an initial material-property space, such as Latin hypercube sampling, simple sequential inhibition process [89], and nonaligned systematic sampling [90]. These methods

produce samples with low discrepancy (dispersed widely within their bounds) while maintaining samples with high diversity (distributed randomly) [90]. It would be interesting to test with new sampling methods and compare areas of their newly generated initial property space. In addition to the two properties studied in this paper, there are many other interesting combinations of properties for various engineering applications, such as porosity and Young's modulus, shear modulus and Young's modulus, etc. The work of Omairey et al. [16] illustrates that their proposed ABAQUS plugin called EasyPBC has a wide range of applications for RVEs with isotropic or anisotropic properties, RVEs in 2D or 3D, and RVEs of homogenous or composite materials [16, 91, 92, 93, 94, 95, 96]. In future works, a 2D material property space is developed by EasyPBC to represent mechanical properties along two different directions of their RVEs, such as E_{11} and E_{22}; ν_{12} and ν_{21}, etc.

3) Material property space exploration under uncertainty: This study aims to provide brief insights into the uncertainty of the combinations of two properties by the visual changes shown as the stochastic boundaries of a material-property space. To this end, the surface distortion is simplified and mimicked as the variations of the geometric parameters. However, a more accurate representation of this defect can be achieved by parameterizing it as the differences in centroid locations, surface curvatures, and the areas of the pore between the printed and simulated structures. In the future, we plan to physically print our CMMs as thin structural panels and construct this stochastic representation based on the experimental data of our printed models. Similar to the surface distortion, uncertainty arising from any other manufacturing defects can be efficiently quantified by the Monte Carlo sampling method facilitated by deep learning with fully connected layers as long as the defects can be parameterized [97]. However, in some cases, it may be difficult to parameterize the defects in manufacturing. In such cases, convolutional neural networks (CNNs) can be used to detect manufacturing defects by extracting features from the given images [98, 99, 100]. With the

application of CNNs, the stochastic boundaries of material property spaces associated with more complex manufacturing defects can also be determined.

4) Inverse design framework with the CGAN: Although the results of this study are based on a specific dataset for spinodoid metamaterials with a particular porosity, the same inverse design framework of spinodoid metamaterials can be extended to design other metamaterials with microstructures represented by binary images (e.g. porous metamaterials [101], truss metamaterials [102], and chiral metamaterials [103]). Considering the wide application of spinodoid metamaterials [33, 34], the design framework is not limited to ULTEM 1000 and its base material can be replaced with various materials for specific applications, such as ultra-high molecular weight polyethylene (UHMWPE) for bone replacement [104] and Polylactic acid (PLA) for energy absorption [105].

5) Concurrent topology optimization method: To overcome the highly compliant behaviors in the optimal designs, future studies can increase the freedom in materials design. For instance, future work may allow geometric patterns of porous metamaterials (microstructures) to vary from the discretized elements to elements in the macrostructures. The design freedom can be further increased by selecting anisotropic metamaterials, such as spinodoid metamaterials. By controlling the orientations of anisotropic materials, this study can further enlarge the design space of the multi-scale structure. Besides, by considering the computational expense of the sensitivity analysis, automatic differentiation can be applied in future studies. The DNN-based surrogate model accelerates the property determination by bypassing the simulation and beneficially provides the sensitivities of its input design variables, which can be easily computed by back-propagating the gradients from the stiffness (outputs of DNN) to the design variables (inputs of DNN) via the chain rule for differentiation [85].

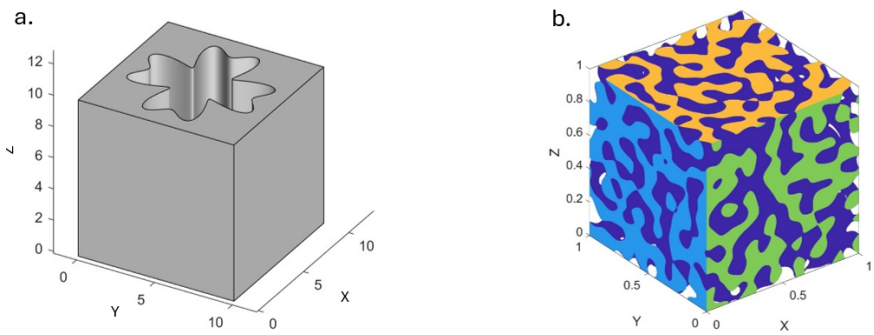


Figure 8.1: Sample 3D RVEs of (a) porous metamaterials and (b) spinodoid metamaterials generated by the GibbonCode[107].

6) Dimension of RVEs: The present study primarily focuses on two classes of cellular mechanical metamaterials in 2D. To retrieve a further accurate analysis of these materials, future studies need to consider extending their 2D RVE models to 3D. The sample 3D RVE models of these materials are shown in Fig. 8.1.

7) Nonlinear mechanical properties: Currently, this study mainly focuses on the linear elastic properties of CMMs under the assumption of small deformations. Accordingly, the displacement of RVEs should be small enough such that the changes in the stiffness of the material due to the loading can be neglected [106]. To determine the nonlinear properties of the CMMs, uniaxial strain tests with a large deformation causing the materials to display plastic or hyper-elastic behavior should be considered [46]. In that case, the effective properties of RVEs are no longer constant like the materials within the elastic region, and they vary based on the prescribed boundary condition (specifically uniaxial stains applied on the surface of RVEs) [46]. In the future, our presented material property space framework can be integrated into nonlinear property calculation by utilizing specific boundary conditions.

Bibliography

[1] Hu, Z., Thiyagarajan, K., Bhusal, A., Letcher, T., Fan, Q. H., Liu, Q., & Salem, D. (2017). Design of ultra-lightweight and high-strength cellular structural composites inspired by biomimetics. In Composites Part B: Engineering (Vol. 121, pp. 108–121). Elsevier BV. https://doi.org/10.1016/j.compositesb.2017.03.033

[2] Sairajan, K. K., Aglietti, G. S., & Mani, K. M. (2016). A review of multifunctional structure technology for aerospace applications. In Acta Astronautica (Vol. 120, pp. 30–42). Elsevier BV. https://doi.org/10.1016/j.actaastro.2015.11.024

[3] Meola, C., Boccardi, S., & Carlomagno, G. maria. (2017). Composite Materials in the Aeronautical Industry. In Infrared Thermography in the Evaluation of Aerospace Composite Materials (pp. 1–24). Elsevier. https://doi.org/10.1016/b978-1-78242-171-9.00001-2

[4] Su, L., Liu, H., Yao, G., & Zhang, J. (2019). Experimental study on the closed-cell aluminum foam shock absorption layer of a high-speed railway tunnel. In Soil Dynamics and Earthquake Engineering (Vol. 119, pp. 331–345). Elsevier BV. https://doi.org/10.1016/j.soildyn.2019.01.012

[5] Yao, S., Xiao, X., Xu, P., Qu, Q., & Che, Q. (2018). The impact performance of honeycomb-filled structures under eccentric loading for subway vehicles. In Thin-Walled Structures (Vol. 123, pp. 360–370). Elsevier BV. https://doi.org/10.1016/j.tws.2017.10.031

[6] Zheng, X., Fu, Z., Du, K., Wang, C., & Yi, Y. (2018). Minimal surface designs for porous materials: from microstructures to mechanical properties. In Journal of Mate-

rials Science (Vol. 53, Issue 14, pp. 10194–10208). Springer Science and Business Media LLC. https://doi.org/10.1007/s10853-018-2285-5

[7] Christensen, J., Kadic, M., Wegener, M., Kraft, O., & Wegener, M. (2015). Vibrant times for mechanical metamaterials. In MRS Communications (Vol. 5, Issue 3, pp. 453–462). Springer Science and Business Media LLC. https://doi.org/10.1557/mrc.2015.51

[8] Zhao, J., Li, Y., & Liu, W. K. (2015). Predicting band structure of 3D mechanical metamaterials with complex geometry via XFEM. In Computational Mechanics (Vol. 55, Issue 4, pp. 659–672). Springer Science and Business Media LLC. https://doi.org/10.1007/s00466-015-1129-2

[9] Wang, Y., Li, F., Wang, Y., Kishimoto, K., & Huang, W. (2008). Tuning of band gaps for a two-dimensional piezoelectric phononic crystal with a rectangular lattice. In Acta Mechanica Sinica (Vol. 25, Issue 1, pp. 65–71). Springer Science and Business Media LLC. https://doi.org/10.1007/s10409-008-0191-9

[10] Chan, Y.-C., Ahmed, F., Wang, L., & Chen, W. (2020). METASET: Exploring Shape and Property Spaces for Data-Driven Metamaterials Design. In Journal of Mechanical Design (Vol. 143, Issue 3). ASME International. https://doi.org/10.1115/1.4048629

[11] Zhou, X., Xiao, Q., & Wang, H. (2022). Metamaterials Design Method based on Deep learning Database. In Journal of Physics: Conference Series (Vol. 2185, Issue 1, p. 012023). IOP Publishing. https://doi.org/10.1088/1742-6596/2185/1/012023

[12] Dai, D., Tang, C., Zhao, D., Xia, S., & Bai, H. (2020). Understanding a Deep Neural Network Based on Neural-Path Coding. In IEEE Access (Vol. 8, pp. 174495–174506). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/access.2020.3024959

[13] Maaranen, H., Miettinen, K., & Mäkelä, M. M. (2004). Quasi-random initial population for genetic algorithms. In Computers amp; Mathematics with Applications (Vol. 47, Issue 12, pp. 1885–1895). Elsevier BV. https://doi.org/10.1016/j.camwa.2003.07.011

[14] Maaranen, H., Miettinen, K., & Penttinen, A. (2006). On initial populations of a genetic algorithm for continuous optimization problems. In Journal of Global Optimization (Vol. 37, Issue 3, pp. 405–436). Springer Science and Business Media LLC. https://doi.org/10.1007/s10898-006-9056-6

[15] López, F. "TPOT: Pipelines Optimization with Genetic Algorithms." Medium. Towards Data Science, December 21, 2020. URL https://towardsdatascience.com/tpot-pipelines-optimization-with-genetic-algorithms-56ec44ef6ede

[16] Omairey, S. L., Dunning, P. D., & Sriramula, S. (2018). Development of an ABAQUS plugin tool for periodic RVE homogenisation. In Engineering with Computers (Vol. 35, Issue 2, pp. 567–577). Springer Science and Business Media LLC. https://doi.org/10.1007/s00366-018-0616-4

[17] Ashby, M.F., Materials Selection in Mechanical Design, 5 edition, ButterworthHeinemann, Cambridge, MA, 2016.

[18] Hashin, Z., & Shtrikman, S. (1962). A Variational Approach to the Theory of the Effective Magnetic Permeability of Multiphase Materials. In Journal of Applied Physics (Vol. 33, Issue 10, pp. 3125–3131). AIP Publishing. https://doi.org/10.1063/1.1728579

[19] Voigt, W., Lehrbuch der Kristallphysik, B.G. Teubner Verlag, Leipzig, 1928.

[20] Reuss, A. (1929) Berechnung der Fliessgrenze von Mischkristallen auf Grund der Plastizitätsbedingung für Einkristalle. ZAMM—Journal of Applied Mathematics

and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, 9, 49-58. http://dx.doi.org/10.1002/zamm.19290090104

[21] Adams, B.L., Kalidindi, S.R., Fullwood, D.T., Microstructure-sensitive Design for Performance Optimization, 1 edition, Butterworth-Heinemann, 2012.

[22] Fast, T., Knezevic, M., & Kalidindi, S. R. (2008). Application of microstructure sensitive design to structural components produced from hexagonal polycrystalline metals. In Computational Materials Science (Vol. 43, Issue 2, pp. 374–383). Elsevier BV. https://doi.org/10.1016/j.commatsci.2007.12.002

[23] PROUST, G., & KALIDINDI, S. (2006). Procedures for construction of anisotropic elastic–plastic property closures for face-centered cubic polycrystals using first-order bounding relations. In Journal of the Mechanics and Physics of Solids (Vol. 54, Issue 8, pp. 1744–1762). Elsevier BV. https://doi.org/10.1016/j.jmps.2006.01.010

[24] Kim, T., Elango, P., Malyuta, D., & Acikmese, B. (2022). Guided Policy Search using Sequential Convex Programming for Initialization of Trajectory Optimization Algorithms. In 2022 American Control Conference (ACC). 2022 American Control Conference (ACC). IEEE. https://doi.org/10.23919/acc53348.2022.9867151

[25] Li, J., Guo, H., Sun, P., Wang, Y., Huang, S., Yuan, T., & Zhang, H. (2022). Topology optimization of anisotropy hierarchical honeycomb acoustic metamaterials for extreme multi-broad band gaps. In Mechanics of Advanced Materials and Structures (pp. 1–13). Informa UK Limited. https://doi.org/10.1080/15376494.2022.2079027

[26] Meng, H., Chronopoulos, D., Fabro, A. T., Maskery, I.,& Chen, Y. (2020). Optimal design of rainbow elastic metamaterials. In International Journal of Mechanical Sciences (Vol. 165, p. 105185). Elsevier BV. https://doi.org/10.1016/j.ijmecsci.2019.105185

[27] Jian, Y., Tang, L., Hu, G., Wang, Y., & Aw, K. C. (2022). Adaptive genetic algorithm enabled tailoring of piezoelectric metamaterials for optimal vibration attenuation. In Smart Materials and Structures (Vol. 31, Issue 7, p. 075026). IOP Publishing. https://doi.org/10.1088/1361-665x/ac775d

[28] Dos Reis, F., & Karathanasopoulos, N. (2022). Inverse metamaterial design combining genetic algorithms with asymptotic homogenization schemes. In International Journal of Solids and Structures (Vol. 250, p. 111702). Elsevier BV. https://doi.org/10.1016/j.ijsolstr.2022.111702

[29] Luo, P., Lan, G., Nong, J., Zhang, X., Xu, T., & Wei, W. (2022). Broadband coherent perfect absorption employing an inverse-designed metasurface via genetic algorithm. In Optics Express (Vol. 30, Issue 19, p. 34429). Optica Publishing Group. https://doi.org/10.1364/oe.468842

[30] Liu, S., & Acar, P. (2022). Inverse Multiscale Design of Cellular Mechanical Metamaterials. In AIAA SCITECH 2022 Forum. AIAA SCITECH 2022 Forum. American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2022-1135

[31] Kumar, A., Datta, S., Roy, T., & Mahapatra, S. S. (2022). Parametric studies of fused filament fabrication towards fabrication of 2D auxetic metamaterial cellular structure followed by auxeticity simulation. In Sādhanā (Vol. 47, Issue 4). Springer Science and Business Media LLC. https://doi.org/10.1007/s12046-022-01993-1

[32] Yadav, A., Rohru, P., Babbar, A., Kumar, R., Ranjan, N., Chohan, J. S., Kumar, R., & Gupta, M. (2022). Fused filament fabrication: A state-of-the-art review of the technology, materials, properties and defects. In International Journal on Interactive Design and Manufacturing (IJIDeM). Springer Science and Business Media LLC. https://doi.org/10.1007/s12008-022-01026-5

[33] Kumar, S., Tan, S., Zheng, L., & Kochmann, D. M. (2020). Inverse-designed spinodoid metamaterials. In npj Computational Materials (Vol. 6, Issue 1). Springer Science and Business Media LLC. https://doi.org/10.1038/s41524-020-0341-6

[34] Röding, M., Wåhlstrand Skärström, V., & Lorén, N. (2022). Inverse design of anisotropic spinodoid materials with prescribed diffusivity. In Scientific Reports (Vol. 12, Issue 1). Springer Science and Business Media LLC. https://doi.org/10.1038/s41598-022-21451-6

[35] du Plessis, A., Broeckhoven, C., Yadroitsava, I., Yadroitsev, I., Hands, C. H., Kunju, R., Bhate, D. (2019). Beautiful and Functional: A Review of Biomimetic Design in Additive Manufacturing. In Additive Manufacturing (Vol. 27, pp. 408–427). Elsevier BV. https://doi.org/10.1016/j.addma.2019.03.033.

[36] Ha, N. S., Lu, G. (2020). A review of recent research on bio-inspired structures and materials for energy absorption applications. In Composites Part B: Engineering (Vol. 181, p. 107496). Elsevier BV. https://doi.org/10.1016/j.compositesb.2019.107496

[37] Wu W, Hu W, Qian G, Liao H, Xu X, Berto F. Mechanical design and multifunctional applications of chiral mechanical metamaterials: A review. Mater Des 2019;180:107950.

[38] Maconachie T, Leary M, Lozanovski B, Zhang X, Qian M, Faruque O, Brandt M. SLM lattice structures: Properties, performance, applications and challenges. Mater Des 2019;183:108137.

[39] Liu, H., Chen, L., Jiang, Y., Zhu, D., Zhou, Y., Wang, X. (2023). Multiscale optimization of additively manufactured graded non-stochastic and stochastic lattice structures. In Composite Structures (Vol. 305, p. 116546). Elsevier BV. https://doi.org/10.1016/j.compstruct.2022.116546

[40] Sigmund, O. (2001). A 99 line topology optimization code written in Matlab. In Structural and Multidisciplinary Optimization (Vol. 21, Issue 2, pp. 120–127). Springer Science and Business Media LLC. https://doi.org/10.1007/s001580050176

[41] Andreassen, E., Clausen, A., Schevenels, M., Lazarov, B. S., Sigmund, O. (2010). Efficient topology optimization in MATLAB using 88 lines of code. In Structural and Multidisciplinary Optimization (Vol. 43, Issue 1, pp. 1–16). Springer Science and Business Media LLC. https://doi.org/10.1007/s00158-010-0594-7

[42] Liu, K., Tovar, A. (2014). An efficient 3D topology optimization code written in Matlab. In Structural and Multidisciplinary Optimization (Vol. 50, Issue 6, pp. 1175–1196). Springer Science and Business Media LLC. https://doi.org/10.1007/s00158-014-1107-x

[43] Gao, J., Luo, Z., Xia, L., Gao, L. (2019). Concurrent topology optimization of multiscale composite structures in Matlab. In Structural and Multidisciplinary Optimization (Vol. 60, Issue 6, pp. 2621–2651). Springer Science and Business Media LLC. https://doi.org/10.1007/s00158-019-02323-6

[44] Liu, S., & Acar, P. (2023). Parameter Space Exploration of Cellular Mechanical Metamaterials Using Genetic Algorithms. In AIAA Journal (Vol. 61, Issue 8, pp. 3633–3643). American Institute of Aeronautics and Astronautics (AIAA). https://doi.org/10.2514/1.j062864

[45] Terada, K., Hori, M., Kyoya, T., & Kikuchi, N. (2000). Simulation of the multiscale convergence in computational homogenization approaches. In International Journal of Solids and Structures (Vol. 37, Issue 16, pp. 2285–2311). Elsevier BV. doi: 10.1016/s0020-7683(98)00341-2

[46] Xue, T., Beatson, A., Chiaramonte, M., Roeder, G., Ash, J. T., Menguc, Y., Adriaenssens, S., Adams, R. P., & Mao, S. (2020). A data-driven computational scheme for the nonlinear mechanical properties of cellular mechanical metamaterials under large deformation. In Soft Matter (Vol. 16, Issue 32, pp. 7524–7534). Royal Society of Chemistry (RSC). https://doi.org/10.1039/d0sm00488j

[47] Cahn, J. W., & Hilliard, J. E. (1958). Free Energy of a Nonuniform System. I. Interfacial Free Energy. In The Journal of Chemical Physics (Vol. 28, Issue 2, pp. 258–267). AIP Publishing. https://doi.org/10.1063/1.1744102

[48] Soyarslan, C., Bargmann, S., Pradas, M., & Weissmüller, J. (2018). 3D stochastic bicontinuous microstructures: Generation, topology and elasticity. In Acta Materialia (Vol. 149, pp. 326–340). Elsevier BV. https://doi.org/10.1016/j.actamat.2018.01.005

[49] Gao, J., Li, H., Luo, Z., Gao, L., & Li, P. (2019). Topology Optimization of Micro-Structured Materials Featured with the Specific Mechanical Properties. In International Journal of Computational Methods (Vol. 17, Issue 03, p. 1850144). World Scientific Pub Co Pte Lt. doi: 10.1142/s021987621850144x

[50] Xia, L., & Breitkopf, P. (2015). Design of materials using topology optimization and energy-based homogenization approach in Matlab. In Structural and Multidisciplinary Optimization (Vol. 52, Issue 6, pp. 1229–1241). Springer Science and Business Media LLC. https://doi.org/10.1007/s00158-015-1294-0

[51] Gao, J., Li, H., Gao, L., Xiao, M. (2018). Topological shape optimization of 3D micro-structured materials using energy-based homogenization method. In Advances in Engineering Software (Vol. 116, pp. 89–102). Elsevier BV. https://doi.org/10.1016/j.advengsoft.2017.12.002

[52] Gopinath, G., Batra, R. C. (2022). Effect of boundary conditions on values of homogenized fabrics's elastic parameters and inelastic response. In Composite Structures (Vol. 301, p. 116229). Elsevier BV. https://doi.org/10.1016/j.compstruct.2022.116229

[53] Lambora, A., Gupta, K., Chopra, K. (2019). Genetic Algorithm- A Literature Review. In 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). IEEE. https://doi.org/10.1109/comitcon.2019.8862255

[54] Chadha, A., Sharma, S., Arora, V. (Eds.). (2024). Computational Science and Its Applications (1st ed.). Apple Academic Press. https://doi.org/10.1201/9781003347484

[55] Tung, L. V., Ha-Van, N., & Seo, C. (2020). A Novel Metamaterial Design Using Genetic Algorithm for High Gain Energy Harvesting Antenna. In 2020 IEEE Wireless Power Transfer Conference (WPTC). 2020 IEEE Wireless Power Transfer Conference (WPTC). IEEE. https://doi.org/10.1109/wptc48563.2020.9295587

[56] Chen, P. Y., Chen, C. H., Wang, H., Tsai, J. H., & Ni, W. X. (2008). Synthesis design of artificial magnetic metamaterials using a genetic algorithm. In Optics Express (Vol. 16, Issue 17, p. 12806). Optica Publishing Group. https://doi.org/10.1364/oe.16.012806

[57] McKay, M. D., Beckman, R. J., & Conover, W. J. (1979). Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. In Technometrics (Vol. 21, Issue 2, pp. 239–245). Informa UK Limited. https://doi.org/10.1080/00401706.1979.10489755

[58] Shahbazi, Z., & Byun, Y.-C. (2022). An Optimized Ensemble Model for Prediction the Bandwidth of Metamaterial Antenna. In Computers, Materials & Continua (Vol.

71, Issue 1, pp. 199–213). Computers, Materials and Continua (Tech Science Press). https://doi.org/10.32604/cmc.2022.021886

[59] Gholamalinezhad, H., Khosravi, H. (2020). Pooling Methods in Deep Neural Networks, a Review (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2009.07485

[60] O'Shea, K., Nash, R. (2015). An Introduction to Convolutional Neural Networks (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1511.08458

[61]

[62] Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-Image Translation with Conditional Adversarial Networks. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE. https://doi.org/10.1109/cvpr.2017.632

[63] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative Adversarial Networks: An Overview. In IEEE Signal Processing Magazine (Vol. 35, Issue 1, pp. 53–65). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/msp.2017.2765202

[64] Lee, M., & Seok, J. (2017). Controllable Generative Adversarial Network (Version 5). arXiv. https://doi.org/10.48550/ARXIV.1708.00598

[65] Kouznetsova, V., Brekelmans, W. A. M., & Baaijens, F. P. T. (2001). An approach to micro-macro modeling of heterogeneous materials. In Computational Mechanics (Vol. 27, Issue 1, pp. 37–48). Springer Science and Business Media LLC. https://doi.org/10.1007/s004660000212

[66] "Boundary of a set of points in 2-D or 3-D." MATLAB.

https://www.mathworks.com/help/matlab/ref/boundary.html. Accessed May 30, 2023.

[67] SABIC Innovative Plastics ULTEM 1000R PEI, http://www.lookpolymers.com/polymer_SABIC-Innovative-Plastics-ULTEM-1000R-PEI.php. Accessed March 18, 2020.

[68] Jahan, A., Edwards, K. L., & Bahraminasab, M. (2016). Multi-criteria decision analysis: For supporting the selection of engineering materials in product design. Butterworth-Heinemann.

[69] Shercliff, H. R., & Ashby, M. F. (2016). Elastic Structures in Design. In Reference Module in Materials Science and Materials Engineering. Elsevier. https://doi.org/10.1016/b978-0-12-803581-8.02944-1

[70] Alghamdy, M., Ahmad, R., & Alsayyed, B. (2019). Material Selection Methodology for Additive Manufacturing Applications. In Procedia CIRP (Vol. 84, pp. 486–490). Elsevier BV. https://doi.org/10.1016/j.procir.2019.04.265

[71] Röding, M., Wåhlstrand Skärström, V., & Lorén, N. (2022). Inverse design of anisotropic spinodoid materials with prescribed diffusivity. In Scientific Reports (Vol. 12, Issue 1). Springer Science and Business Media LLC. https://doi.org/10.1038/s41598-022-21451-6

[72] Zheng, X., Chen, T.-T., Guo, X., Samitsu, S., & Watanabe, I. (2021). Controllable inverse design of auxetic metamaterials using deep learning. In Materials & Design (Vol. 211, p. 110178). Elsevier BV. https://doi.org/10.1016/j.matdes.2021.110178

[73] Wickramasinghe, S., Do, T., & Tran, P. (2020). FDM-Based 3D Printing of Polymer and Associated Composite: A Review on Mechanical Properties, De-

fects and Treatments. In Polymers (Vol. 12, Issue 7, p. 1529). MDPI AG. https://doi.org/10.3390/polym12071529

[74] Butt, J., & Bhaskar, R. (2020). Investigating the Effects of Annealing on the Mechanical Properties of FFF-Printed Thermoplastics. In Journal of Manufacturing and Materials Processing (Vol. 4, Issue 2, p. 38). MDPI AG. https://doi.org/10.3390/jmmp4020038

[75] Sunny, S., Chen, H., Malik, A., & Lu, H. (2021). Influence of residual stress and fluid–structure interaction on the impact behavior of fused filament fabrication components. In Additive Manufacturing (Vol. 37, p. 101662). Elsevier BV. https://doi.org/10.1016/j.addma.2020.101662

[76] "Accuracies and Tolerances in 3D Printing." 3Faktur, November 12, 2019. https://3faktur.com/en/accuracies-and-tolerances-in-3d-printing/. Accessed June 3, 2023.

[77] Minetola, P., Calignano, F., & Galati, M. (2020). Comparing geometric tolerance capabilities of additive manufacturing systems for polymers. In Additive Manufacturing (Vol. 32, p. 101103). Elsevier BV. https://doi.org/10.1016/j.addma.2020.101103

[78] Haase, P. (1995). Spatial pattern analysis in ecology based on Ripley's K-function: Introduction and methods of edge correction. In Journal of Vegetation Science (Vol. 6, Issue 4, pp. 575–582). Wiley. https://doi.org/10.2307/3236356

[79] Gurbuz, C., Kronowetter, F., Dietz, C., Eser, M., Schmid, J., & Marburg, S. (2021). Generative adversarial networks for the design of acoustic metamaterials. In The Journal of the Acoustical Society of America (Vol. 149, Issue 2, pp. 1162–1174). Acoustical Society of America (ASA). https://doi.org/10.1121/10.0003501

[80] Zheng, X., Chen, T.-T., Guo, X., Samitsu, S., & Watanabe, I. (2021). Controllable inverse design of auxetic metamaterials using deep learning. In Materials & Design (Vol. 211, p. 110178). Elsevier BV. https://doi.org/10.1016/j.matdes.2021.110178

[81] Agarap, A. F. (2018). Deep Learning using Rectified Linear Units (ReLU) (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1803.08375

[82] Alomarah, A., Masood, S. H., & Ruan, D. (2022). Metamaterials with enhanced mechanical properties and tuneable Poisson's ratio. In Smart Materials and Structures (Vol. 31, Issue 2, p. 025026). IOP Publishing. https://doi.org/10.1088/1361-665x/ac3c08

[83] Bendsøe, M. P. (1989). Optimal shape design as a material distribution problem. In Structural Optimization (Vol. 1, Issue 4, pp. 193–202). Springer Science and Business Media LLC. https://doi.org/10.1007/bf01650949

[84] Zhou, M., Rozvany, G. I. N. (1991). The COC algorithm, Part II: Topological, geometrical and generalized shape optimization. In Computer Methods in Applied Mechanics and Engineering (Vol. 89, Issues 1–3, pp. 309–336). Elsevier BV. https://doi.org/10.1016/0045-7825(91)90046-9

[85] Zheng, L., Kumar, S., Kochmann, D. M. (2021). Data-driven topology optimization of spinodoid metamaterials with seamlessly tunable anisotropy. In Computer Methods in Applied Mechanics and Engineering (Vol. 383, p. 113894). Elsevier BV. https://doi.org/10.1016/j.cma.2021.113894

[86] Brown, B., Hmeidat, N. S., Jia, X., Wilt, J., Roberts, M., Compton, B. G., & Vermaak, N. (2022). Experimental investigations of the effectiveness of simultaneous topology/orientation optimization via SOMP and princi-

pal stress directions. In Materials & Design (Vol. 217, p. 110647). Elsevier BV. https://doi.org/10.1016/j.matdes.2022.110647

[87] O, B. J., M. O, O., C.E, O. (2017). Evaluating Linear Second Order Homogenous Differential Equations Using the Finite Difference Schemes. In IOSR Journal of Mathematics (Vol. 13, Issue 03, pp. 43–47). IOSR Journals. https://doi.org/10.9790/5728-1303044347

[88] Hoffman, J. D. (2001). Numerical methods for engineers and scientists: Joe D. Hoffman. CRC Press.

[89] Dekkers, A., & Aarts, E. (1991). Global optimization and simulated annealing. In Mathematical Programming (Vol. 50, Issues 1–3, pp. 367–393). Springer Science and Business Media LLC. https://doi.org/10.1007/bf01594945

[90] Maaranen, H., Miettinen, K. & Penttinen, A. On initial populations of a genetic algorithm for continuous optimization problems. J Glob Optim 37, 405–436 (2007). https://doi.org/10.1007/s10898-006-9056-6

[91] Fong, E., Omairey, S. L., & Dunning, P. D. (2020). Design of multifunctional metamaterials using optimization (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2004.13571

[92] Orlova, D. A., Panchenko, A. Yu., Omairey, S. L., & Berinskii, I. E. (2022). Computational homogenization of bio-inspired metamaterial with a random fiber network microstructure. In Mechanics Research Communications (Vol. 124, p. 103930). Elsevier BV. https://doi.org/10.1016/j.mechrescom.2022.103930

[93] Sangsefidi, A. R., Dibajian, S. H., Kadkhodapour, J., Anaraki, A. P., Schmauder, S., & Schneider, Y. (2021). An Abaqus plugin for evaluation of the Auxetic structure per-

formance. In Engineering with Computers (Vol. 38, Issue S2, pp. 1681–1704). Springer Science and Business Media LLC. https://doi.org/10.1007/s00366-021-01295-w

[94] Zhao, Y., Wang, Y., Hao, J., Wang, Y., Wang, K., & Tai, S. (2023). Study on mechanical properties of cellular structures with negative Poisson's ratio based on the development of Abaqus plug-in tool. In Composite Structures (Vol. 322, p. 117348). Elsevier BV. https://doi.org/10.1016/j.compstruct.2023.117348

[95] Schwahofer, O., Büttner, S., Binder, J., Colin, D., & Drechsler, K. (2022). Multiscale Optimization of 3D-Printed Beam-Based Lattice Structures through Elastically Tailored Unit Cells. In Advanced Engineering Materials (Vol. 25, Issue 20). Wiley. https://doi.org/10.1002/adem.202201385

[96] Zhu, L., Wang, X., Sun, L., Hu, Q., & Li, N. (2022). Optimisation of Selective Laser Melted Ti6Al4V Functionally Graded Lattice Structures Accounting for Structural Safety. In Materials (Vol. 15, Issue 24, p. 9072). MDPI AG. https://doi.org/10.3390/ma15249072

[97] Yang, J., Li, S., Wang, Z., Dong, H., Wang, J., & Tang, S. (2020). Using Deep Learning to Detect Defects in Manufacturing: A Comprehensive Survey and Current Challenges. In Materials (Vol. 13, Issue 24, p. 5755). MDPI AG. https://doi.org/10.3390/ma13245755

[98] Perez, H., Tah, J. H. M., & Mosavi, A. (2019). Deep Learning for Detecting Building Defects Using Convolutional Neural Networks. In Sensors (Vol. 19, Issue 16, p. 3556). MDPI AG. https://doi.org/10.3390/s19163556

[99] Dimitriou, N., Leontaris, L., Vafeiadis, T., Ioannidis, D., Wotherspoon, T., Tinker, G., & Tzovaras, D. (2020). A Deep Learning framework for simulation and defect

prediction applied in microelectronics. In Simulation Modelling Practice and Theory (Vol. 100, p. 102063). Elsevier BV. https://doi.org/10.1016/j.simpat.2019.102063

[100] Yang, J., Li, S., Wang, Z., & Yang, G. (2019). Real-Time Tiny Part Defect Detection System in Manufacturing Using Deep Learning. In IEEE Access (Vol. 7, pp. 89278–89291). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/access.2019.2925561

[101] Librandi, G., Moshe, M., Lahini, Y., & Bertoldi, K. (2017). Porous mechanical metamaterials as interacting elastic charges (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1709.00328

[102] Glaesener, R. N., Bastek, J.-H., Gonon, F., Kannan, V., Telgen, B., Spöttling, B., Steiner, S., & Kochmann, D. M. (2021). Viscoelastic truss metamaterials as time-dependent generalized continua. In Journal of the Mechanics and Physics of Solids (Vol. 156, p. 104569). Elsevier BV. https://doi.org/10.1016/j.jmps.2021.104569

[103] Wu, W., Hu, W., Qian, G., Liao, H., Xu, X., & Berto, F. (2019). Mechanical design and multifunctional applications of chiral mechanical metamaterials: A review. In Materials & Design (Vol. 180, p. 107950). Elsevier BV. https://doi.org/10.1016/j.matdes.2019.107950

[104] Senra, M. R., & Marques, M. de F. V. (2020). Synthetic Polymeric Materials for Bone Replacement. In Journal of Composites Science (Vol. 4, Issue 4, p. 191). MDPI AG. https://doi.org/10.3390/jcs4040191

[105] Yazdani Sarvestani, H., Akbarzadeh, A. H., Niknam, H., & Hermenean, K. (2018). 3D printed architected polymeric sandwich panels: Energy absorption and structural performance. In Composite Structures (Vol. 200, pp. 886–909). Elsevier BV. https://doi.org/10.1016/j.compstruct.2018.04.002

[106] Wenz, F., Schmidt, I., Leichner, A., Lichti, T., Baumann, S., Andrae, H., & Eberl, C. (2021). Designing Shape Morphing Behavior through Local Programming of Mechanical Metamaterials. In Advanced Materials (Vol. 33, Issue 37). Wiley. https://doi.org/10.1002/adma.202008617

[107] Moerman, K. M. (2019). GIBBON. GitHub repository. https://github.com/gibbonCode/GIBBON

Appendices

Appendix A

Figures: Converging history of the sensitivities of the homogenized stiffness of porous metamaterials

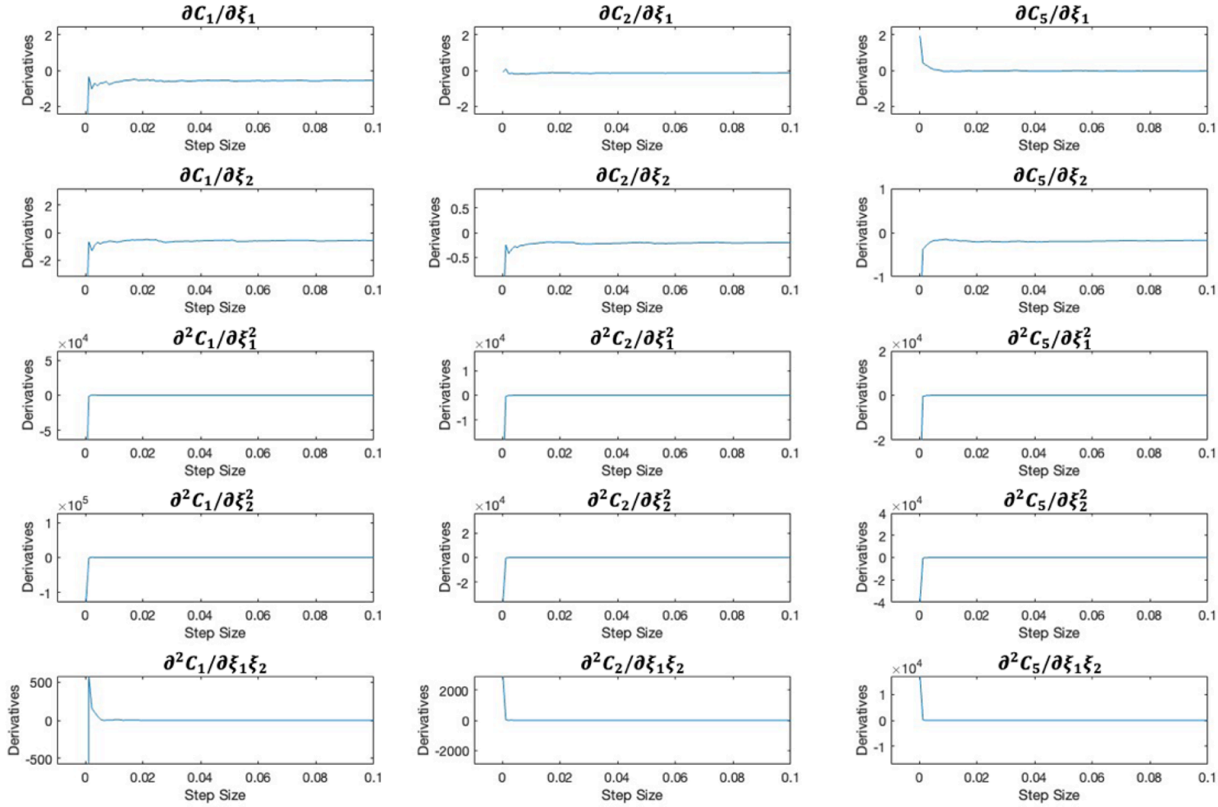


Figure A.1: Converging history of the first and second derivatives of the homogenized stiffness of porous metamaterials that varies with different step sizes (microstructures in the dimensions of 128 x 128).

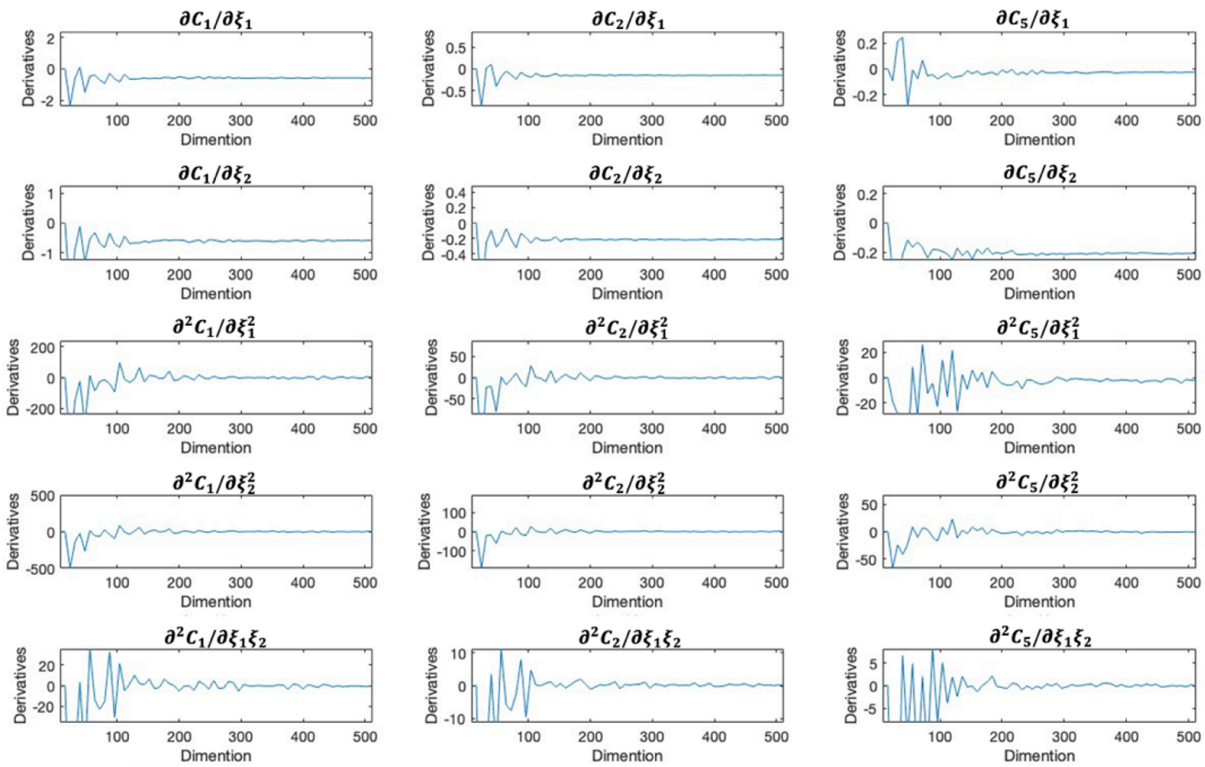


Figure A.2: Converging history of the first and second derivatives of the homogenized stiffness of porous metamaterials that varies with their different dimensions.